DRAFT CURRICULUM-2026

(C-26)

DIPLOMA IN COMUPTER ENGINEERING

STATE BOARD OF TECHNICAL EDUCATION AND TRAINING, ANDHRA PRADESH

FIRST YEAR								
THEORY SUBJECTS	COMPULSORY SUBJECTS	**S No**	**Sub. Code**	**Subject**	**PRACTICUM (Y/N)**	**Periods**	**Total Periods**	**Credits**
		1	26CM101T	ENGLISH COMMUNICATION	N	3	90	4
		2	26CM102T	ENGG MATHEMATICS	N	6	180	6
		3	26CM103T	ENGG PHYSICS	N	3	90	4
		4	26CM104T	ENGG CHEMISTRY	N	3	90	4
		5	26CM105T	PYTHON PROGRAMMING	N	6	180	6
		AUDIT SUBJECT						
		6	26CM106A	CODING FUNDAMENTALS	Y	2	60	
		TOTAL THEORY + PRACTICUM PERIODS				**23**	**690**	**24**
PRACTICAL SUBJECTS	COMPULSORY SUBJECTS	7	26CM107L	PYTHON PROGRAMMING LAB	N	6	180	6
		8	26CM108L	COMPUTER HARDWARE LAB	Y	4	120	3
		9	26CM109L	PHYSICS AND CHEMISTRY LAB	N	3 (1.5+1.5)	90 (45+45)	6
		10	26CM110L	COMPUTER AND DIGITAL SKILLS LAB	Y	3	90	3
		11		ACTIVITY	…	3	90	1
		TOTAL PRACTICAL + PRACTICUM PERIODS				**19**	**570**	**16**
GRAND TOTAL PERIODS (30 WEEKS PER YEAR)						**42**	**1260**	**40**

Course code	Course Title	No. Of periods/week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM101T	ENGLISH ESSENTIALS	3	90	30	70	3

TIME SCHEDULE

S. No.	Chapter/ Unit Title	No. of Periods	Weightage of marks	No. of Short Answer Questions	No. of Essay Questions	CO's Mapped
1.	Exploring English	10	11	1	1	CO1, CO2, CO3, CO4, CO5
2.	The Better You!	10	11	3	1	CO1, CO2, CO3, CO4, CO5
3.	Drive to Destiny!	10	14		1	CO1, CO2, CO3, CO4, CO5
4.	Renew, Rewire & Resolve!	10	14	1	1	CO1, CO2, CO3, CO4, CO5
5.	Brains & Bots	10		1		CO1, CO2, CO3, CO4, CO5
6	The Blue Planet: Mend or End	10	11	1	1	CO1, CO2, CO3, CO4, CO5
7	One World One Dream	10	11	1	1	CO1, CO2, CO3, CO4, CO5
8	The Net Norms	10	11	1	1	CO1, CO2, CO3, CO4, CO5
9	Managing Moods &	10	11	1	1	CO1, CO2, CO3, CO4,

	Moments					CO5
	Total	**90**	**94**	**10**	**8**	

COURSE OBJECTIVES

(i)	To inculcate knowledge of functional English and enrich vocabulary.
(ii)	To impart effective listening, speaking, reading, and writing skills.
(iii)	To sensitise the students on themes related to personality, technological advancements, sustainability, and human values.

COURSE OUTCOMES:

Upon completion of the course, the student shall be able to:

CO1	CM101.1	Learn and apply various English grammatical concepts to communicate in academic, professional, personal, and social contexts.
CO2	CM101.2	Use appropriate vocabulary in academic, professional, and in business correspondence and on social media platforms.
CO3	CM101.3	Listen and understand, read and comprehend different forms of academic, professional, and general listening and reading materials.
CO4	CM101.4	Communicate effectively and fluently in oral and written forms in various life situations.
CO5	CM101.5	Display scientific temper and universal human values technology for holistic development and harmonious living through demeanour and communication.

LEARNING OUTCOMES

1.0 EXPLORING ENGLISH

1.1 To read and comprehend simple sentences in a short passage.

1.2 To apply certain rules of spelling, correct the misspelt words and use dictionary to enrich vocabulary

1.3 To identify various parts of speech suitable to context and use articles & prepositions accurately.

1.4 To describe a given situation/ picture using simple sentences.

1.5 To value the importance of English for employability.

2.0 THE BETTER YOU!

2.1 TO READ and comprehend formal and informal conversations.

2.2 To use words suitable to the context in spoken and written communication.

2.3 To use the appropriate forms of verbs.

2.4 To engage in conversations in both formal and informal contexts.

2.5 To demonstrate positive attitude in personal and academic spheres.

3.0 DRIVE TO DESTINY

3.1 To read and comprehend paragraphs for specific and general information, and distinguish different types of paragraphs

3.2 To distinguish word pairs and use them contextually.

3. 3 To frame sentences with proper subject-verb agreement.

3.4 To describe actions using appropriate tense.

3.5 To set and achieve academic and personal goals.

4.0 RENEW, REWIRE & RESOLVE

4.1 To read and comprehend the content and structure of e-mails for different purposes.

4.2 To recognize the root words and use appropriate affixes contextually.

4.3 To use various kinds of sentences for different communicative situations.

4.4 To draft E-mails for academic and professional purposes.

4.5 To apply the skills of critical thinking and creativity in solving problems.

5.0 Brains & Bots

5.1 To read and comprehend the description of a process with sequence markers.

5.2 To communicate using phrasal verbs intelligibly.

5.3 To use active and passive voice appropriately.

5.4 To describe processes and procedures using appropriate sentence forms.

55. To appraise the importance and use of robotics and artificial intelligence in human life.

6.0 THE BLUE PLANET: MEND OR END!

6.1 To read and comprehend the content, structure and purpose of formal and informal letters.

6.2 To describe using appropriate forms of adjectives

6.3 To substitute phrases or clauses with a single word.

6.4 To draft personal and professional letters.

6.5 To realise the importance of environmental protection and ensure sustainability.

7.0 ONE WORLD - ONE DREAM

7.1 To read and comprehend essay and analyse its features

7.2 To identify and create shortened forms of words or phrases.

7.3 To report the expressions of the speaker to a third person with necessary grammatical changes.

7.4 To draft well-organized essays for academic and professional purposes.

7.5 To appraise the importance of inclusivity in the society.

8.0 THE NET NORMS

8.1 To comprehend and analyse the given text for making notes and summarising.

8.2 To use contemporary language in informal communication.

8.3 To split or combine ideas using conjunctions for effective communication.

8.4 To make notes of textual information and summarize the information.

8.5 To demonstrate ideal behaviour on the internet.

9.0 MANAGING MOODS & MOMENTS

9.1 To read and comprehend different types of reports.

9.2 To analyse and evaluate grammatical errors.

9.3 To use words and phrases in sentences of your own.

9.4 To draft organized and comprehensive reports on experiments, events, visits and incidents.

9.5 To assess the reasons and manage stress and manage time effectively.

CO-PO/PSO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PS01	PS02	PS03
CO1	PO's 1 to 4 are not directly applicable for English course. However, activities that use content from science and technology relevant to the Programme taken up by the student shall be exploited for communication in the Course.					3	2	Programme Specific Outcomes are Branch specific with technical aspects which are not directly applicable to English Language course.		
CO2						3	2			
CO3						3	2			
CO4						3	2			
CO5					1		2			
Average					1	3	2			

3-Strongly Mapped 2- Moderately Mapped 1- Slightly Mapped

Note: The gaps in CO and PO mapping will be met by one or more appropriate activities from the following:
(i) Assignments (ii) Tutorials (iii) Seminars (iv) Guest Lectures (v) Group Discussions (vi) Quizzes (iv) Library Visits etc.,

COURSE CONTENT

1.0 EXPLORING ENGLISH

Reading – Roleplay – Picture Interpretation – Sounds and Spellings – Parts of Speech – Articles and Prepositions

2.0 THE BETTER YOU!

Reading – Dialogue Writing – Synonyms and Antonyms – Word order – Verbs

3.0 DRIVE TO DESTINY

Reading – Paragraph Writing – Homophones, Homonyms, Homographs – Concord – Tenses

4.0 RENEW, REWIRE & RESOLVE

Reading – E-mail Writing – Roots, Affixes – Kinds of Sentences

5.0 Brains & Bots

Reading – Describing Process – Phrasal Verbs – Voice

6.0 THE BLUE PLANET: MEND OR END!

Reading – Letter Writing – One-word Substitutes – Degrees of Comparison

7.0 ONE WORLD - ONE DREAM

Reading – Essay Writing – Abbreviations & Acronyms – Reported Speech

8.0 THE NET NORMS

Reading – Note making & Summarising – Gen-Z Vocabulary – Synthesis of Sentences

9.0 MANAGING MOODS & MOMENTS

Reading – Report Writing – Usage – Error Analysis

The text book "English Essentials" (A Textbook of English for I Year Engineering Diploma Courses - by SBTET, AP) is the prescribed text for this course. It comprises various language inputs and activities addressing the Learning outcomes specified in each unit. Every unit will have six major components: Listening, Speaking, Reading, Writing, Vocabulary, Grammar. The activities will be designed as Individual, Pair and Group activities to facilitate self and peer learning.

REFERENCE BOOKS

Martin Hewings : *Advanced Grammar in Use,* Cambridge University Press

Murphy, Raymond : *English Grammar in Use,* Cambridge University Press

Sidney Greenbaum : *Oxford English Grammar*, Oxford University Press

Wren and Martin (Revised by N.D.V. Prasad Rao): *English Grammar and Composition*, Blackie ELT Books, S. Chand and Co.

Sarah Freeman : *Strengthen Your Writing*, Macmillan

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TESTS I, II & III

Unit Test	Learning Outcomes to be Covered
Unit Test – 1	From 1.1 to 3.5
Unit Test – 2	From 4.1 to. 6.5
Unit Test – 3	From 7.1 to 9.5

C-26 Common-102

ENGINEERING MATHEMATICS-I

(Common to all Branches)

Course Code	Course Title	No. of Periods/week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM102T	Engineering Mathematics-I	6	180	30	70	6

S.No.	Unit Title	No. of periods	COs mapped
1	Algebra	31	CO1
2	Trigonometry	63	CO2
3	Co-ordinate Geometry	26	CO3
4	Differential Calculus	34	CO4
5	Integral Calculus	26	CO5
Total Periods		**180**	

TIME SCHEDULE

S.No.	Chapter	No. of Periods	Marks Allotted	No. of Short Questions	No. of Essay Questions	COs mapped
Unit - I: Algebra						
1	Partial Fractions	6	4	0	1/2	CO1
2	Matrices and Determinants	25	15	1	1&1/2	CO1
Unit - II: Trigonometry						
3	Trigonometric Ratios	4	0	0	0	CO2
4	Compound Angles	8	3	1	0	CO2
5	Multiple and Sub-multiple angles	8	3	1	0	CO2
6	Transformations	9	4	0	1/2	CO2
7	Inverse Trigonometric Functions	8	4	0	1/2	CO2
8	Trigonometric Equations	8	4	0	1/2	CO2
9	Properties of triangles	8	4	0	1/2	CO2
10	Complex Numbers	8	3	1	0	CO2
11	Hyperbolic functions	2	0	0	0	CO2
Unit III: Co-ordinate Geometry						
12	Straight Lines	8	3	1	0	CO3
13	Circles	8	4	0	1/2	CO3
14	Conic Sections	10	4	0	1/2	CO3
Unit – IV: Differential Calculus						

15	Limits and Continuity	6	3	1	0	CO4
16	Differentiation	28	17	3	1	CO4
Unit – V: Integral Calculus						
17	Indefinite integration	18	11	1	1	CO5
18	Definite integration	8	8	0	1	CO5
	Total	180	94	10	8	
			Marks	30	64	

COURSE OBJECTIVES

Upon completion of the course, the student shall be able to	
(i)	To apply the principles of Algebra, Trigonometry and Co-ordinate Geometry to real-time problems in engineering.
(ii)	To build the concepts of indefinite integrals and definite integrals.

COURSE OUTCOMES

CO1	Resolve partial fractions and solve problems on matrices and determinants.
CO2	Use the concept of trigonometric functions, their inverses and complex numbers.
CO3	Find the equations and properties of straight lines, circles and conic sections in coordinate system.
CO4	Evaluate the limits and derivatives of various functions and apply to engineering problems.
CO5	Integrate various functions using different methods and evaluate definite integrals.

LEARNING OUTCOMES

UNIT - I

C.O. 1 Resolve partial fractions and solve problems on matrices and determinants.

L.O. 1.1 Define rational, proper and improper fractions of polynomials.

1.2 Explain the procedure of resolving proper fractions of the type $\frac{f(x)}{(ax+b)(cx+d)}$

1.3 Define a matrix and order of a matrix.

1.4 State various types of matrices with examples (emphasis on 3rd order square matrices).

1.5 Compute sum, difference, scalar multiplication and product of matrices. Illustrate the properties of these operations such as commutative, associative and distributive properties with examples and counter examples.

1.6 Define the transpose of a matrix and state its properties – examples.

1.7 Define symmetric and skew-symmetric matrices with examples. Resolve a square matrix into a sum of symmetric and skew-symmetric matrices with examples.

1.8 Define determinant of a square matrix; minor, co-factor of an element of a 3x3 square matrix with examples. Expand the determinant of a 3x3 matrix using Laplace expansion formula. State and apply the properties of determinants to solve simple problems.

1.9 Distinguish singular and non-singular matrices. Define multiplicative inverse of a matrix and list properties of adjoint and inverse. Compute adjoint and multiplicative inverse of a square matrix.

1.10 Solve a system of three linear equations in three unknowns using Cramer's rule.

UNIT - II

C.O. 2 Solve problems using the concept of trigonometric functions, their inverses and complex numbers.

L.O. 2.1 Recall the trigonometric ratios and their values at specified angles.

2.2 Draw graphs of trigonometric functions - Explain periodicity of trigonometric functions.

2.3 Define compound angles and state the formulae of sin(A±B), cos(A±B),tan(A±B) and cot(A±B).

2.4 Give simple examples on compound angles to derive the values of $\sin 15^0$,$\cos 15^0$, $\sin 75^0$, $\cos 75^0$, $\tan 15^0$, $\tan 75^0$ etc.

2.5 Derive identities like $\sin(A+B)\sin(A-B) = \sin^2 A - \sin^2 B$ etc.

2.6 Solve simple problems using the identities on compound angles.

2.7 Derive the formulae of multiple angles 2A, 3A etc., and sub-multiple angle A/2 in terms of angle A of trigonometric functions.

2.8 Derive useful allied formulae like $\sin^2 A = (1 - \cos 2A)/2$ etc.

2.9 Solve simple problems using the multiple and sub-multiple formulae.

2.10 Derive the formulae on transforming sum or difference of two trigonometric ratios into a product and vice versa - examples on these formulae.

2.11 Solve problems by applying these formulae to sum or difference or product of two terms.

Syllabus for Unit test-I completed

2.12 Explain the concept of inverse of a trigonometric function by selecting an appropriate domain and range.

2.13 Define inverses of six trigonometric functions along with their domains and ranges.

2.14 Derive relations between inverse trigonometric functions so that the given inverse trigonometric function can be expressed in terms of other inverse trigonometric functions with examples.

2.15 State various properties of inverse trigonometric functions and identities like $\sin^{-1} x + \cos^{-1} x = \frac{\pi}{2}$ etc.

2.16 Apply formulae like $\tan^{-1} x + \tan^{-1} y = \tan^{-1}\left(\frac{x+y}{1-xy}\right)$, *where* $x \geq 0, y \geq 0, xy < 1$ etc., to solve simple problems.

2.17 Explain what is meant by solution of trigonometric equations and find the general solutions of $\sin x = k$, $\cos x = k$ and $\tan x = k$ with appropriate examples.

2.18 Solve models of the type $a\sin^2 x + b\sin x + c = 0$ and $a\sin x + b\cos x = c$.

2.19 State sine rule, cosine rule, tangent rule and projection rule and solve a triangle using these formulae.

2.20 List various formulae for area of a triangle with examples.

2.21 Define a complex number, its modulus, conjugate, amplitude and list their properties.
2.22 Define arithmetic operations on complex numbers with examples.
2.23 Represent the complex number in various forms like modulus-amplitude (polar) form and Exponential (Euler) form with examples.
2.24 Explain the concept of hyperbolic trigonometric functions and list appropriate formulae.

UNIT - III

C.O. 3 Find the equations and properties of straight lines, circles and conic sections in coordinate system.

L.O. 3.1 Write different forms of a straight line – general form, point-slope form, slope- intercept form, two-point form, intercept form and normal form or perpendicular form.
3.2 Find distance of a point from a line, acute angle between two lines, intersection of two non-parallel lines and distance between two parallel lines.
3.3 Define locus of a point and circle.
3.4 Write the general equation of a circle and find its centre and radius.
3.5 Find the equation of a circle, given (i) centre and radius, (ii)two ends of the diameter(iii) three non collinear points of type (0,0) (a,0), (0, b).
3.6 Define a conic - Explain the terms focus, directrix, eccentricity, axes and latus-rectum of a conic.
3.7 Find the equation of a conic when focus, directrix and eccentricity are given.
3.8 Describe the properties of Parabola $y^2 = 4ax$.

Syllabus for Unit test-II completed

UNIT - IV

C.O. 4 Evaluate the limits and derivatives of various functions.

L.O. 4.1 Explain the concept of limit and meaning of $\lim_{x\to a} f(x)=l$ and state the properties of limits.
4.2 Evaluate the limits of the type $\lim_{x\to a}\frac{f(x)}{g(x)}$
4.3 State the Standard limits $\lim_{x\to a}\frac{x^n-a^n}{x-a}$, $\lim_{x\to 0}\frac{\sin x}{x}$, $\lim_{x\to 0}\frac{\tan x}{x}$, $\lim_{x\to 0}\frac{a^x-1}{x}$, $\lim_{x\to 0}\frac{e^x-1}{x}$, **(**without proof) and solve simple problems using these standard limits.
4.4 Explain the concept of continuity of a function at a point and on an interval
4.5 State the concept of derivative of a function y = f(x) – definition, first principle as $\lim_{h\to 0}\frac{f(x+h)-f(x)}{h}$ and also write standard notations to denote the derivative of a function.
4.6 Explain the significance of derivative in scientific and engineering applications.
4.7 Find the derivatives of standard algebraic, logarithmic, exponential and trigonometric functions using the first principle.
4.8 Find the derivatives of hyperbolic and inverse hyperbolic functions.

4.9 State the rules of differentiation of sum, difference, scalar multiplication, product and quotient of functions with simple illustrative examples.

4.10 Explain the method of differentiation of a function of a function (Chain rule) with illustrative examples.

4.11 Explain the method of differentiation of parametric functions with examples.

4.12 Explain the procedure for finding the derivatives of implicit functions with examples.

4.13 Explain the need of taking logarithms for differentiating some functions of $[f(x)]^{g(x)}$ type – examples on logarithmic differentiation.

4.14 Explain the concept of finding the second order derivatives with examples.

4.15 Define maximum and minimum values of a function and find the maximum and minimum values for quadratic polynomials.

4.16 Explain the concept of functions of several variables, finding partial derivatives and difference between the ordinary and partial derivatives with simple examples.

UNIT - V

C.O. 5 Integrate various functions using different methods and evaluate definite integrals.

L.O. 5.1 Explain the concept of Indefinite integral as an anti-derivative.

5.2. State the indefinite integral of standard functions and properties of $\int (u+v)\,dx$ and $\int k\,u\,dx$ where u, v are functions of x and k is constant.

5.3. Solve problems involving standard functions using these properties.

5.4. Evaluate integrals involving simple functions of the following type by the method of substitution.

i) $\int f(ax+b)\,dx$, *where f(x)* is in standard form.

ii) $\int (f(x))^n f'(x)\,dx,\ n \neq -1$

iii) $\int \dfrac{f'(x)}{f(x)}\,dx$

5.5. Find the integrals of *tan x, cot x, sec x* and *cosec x* w.r.t. x.

5.6. Evaluate the Standard integrals of the functions of the type :

$i)\ \dfrac{1}{a^2+x^2}, \dfrac{1}{a^2-x^2}, \dfrac{1}{x^2-a^2}$

$ii)\ \dfrac{1}{\sqrt{a^2+x^2}}, \dfrac{1}{\sqrt{a^2-x^2}}, \dfrac{1}{\sqrt{x^2-a^2}}$

$iii)\ \sqrt{a^2+x^2}, \sqrt{a^2-x^2}, \sqrt{x^2-a^2}$

5.7. Evaluate integrals using decomposition method for integrand of the type $\dfrac{px+q}{(ax+b)(cx+d)}$.

5.8. Solve problems using integration by parts.

5.9 Use Bernoulli's rule to evaluate the integrals of the form $\int u.v\,dx$.

5.10. State the fundamental theorem of integral calculus.

5.11. Explain the concept of definite integral.

5.12. Solve simple problems on definite integrals.

5.13. State various properties of definite integrals.

5.14. Evaluate simple problems on definite integrals using these properties.

Syllabus for Unit test-III completed

CO/PO – MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1	3	2	2	3				3	2	2
CO2	3	3	2	2				3	2	2
CO3	3	3	2	2				3	2	2
CO4	3	3	3	3				3	3	3
CO5	3	3	3	3				3	3	3
Avg.	3	2.8	2.4	2.6				3	2.4	2.4

3 = Strongly mapped (High), **2 =** moderately mapped (Medium), **1 =** slightly mapped (Low)

Note: The gaps in CO/PO mapping can be met with appropriate activities as follows:

For PO5: Appropriate quiz programmes may be conducted at intervals and duration as decided by concerned faculty.
For PO6: Seminars on applications of mathematics in various engineering disciplines are to be planned and conducted.
For PO7: Plan activities in such a way that students can visit the Library to refer standard books on Mathematics and access the latest updates in reputed national and international journals. Additionally, encourage them to attend seminars and learn mathematical software tools.

COURSE CONTENT

Unit-I: Algebra

1. **Partial Fractions:** Definitions of rational, proper and improper fractions of polynomials. Resolve rational fractions (proper fractions) of type $\frac{f(x)}{(ax+b)(cx+d)}$ into partial fractions.
2. **Matrices:** Definition of a matrix, types of matrices - Algebra of matrices, equality of two matrices, sum, difference, scalar multiplication and product of matrices. Transpose of a matrix, Symmetric, skew-symmetric matrices - Determinant of a square matrix, minor and cofactor of an element, Laplace's expansion, properties of determinants - Singular and non-singular matrices, Adjoint and multiplicative inverse of a square matrix - System of linear equations in 3 variables-Solutions by Cramer's rule.

Unit-II: Trigonometry

3. **Trigonometric ratios:** Definition of trigonometric ratios of any angle, values of trigonometric ratios at specified values, draw graphs of trigonometric functions, periodicity of trigonometric functions.
4. **Compound angles:** Formulas of sin(A±B), cos(A±B), tan(A±B), cot(A±B), and related identities.
5. **Multiple and sub-multiple angles:** Formulae for trigonometric ratios of multiple angles 2A, 3A and sub multiple angle A/2.
6. **Transformations:** Transformations of products into sums or differences and vice versa.
7. **Inverse trigonometric functions:** Definition, domains and ranges-basic properties.
8. **Trigonometric equations:** Concept of a solution, principal value and general solution of trigonometric equations:
   Sin x = k, cos x= k, tan x = k, where k is a constant. Solutions of simple quadratic equations and equations of type a sin x + b cos x=c.

9. **Properties of triangles:** Relations between sides and angles of a triangle- sine rule, cosine rule, tangent rule and projection rule-area of a triangle.
10. **Complex Numbers:** Definition of a complex number, modulus, conjugate and amplitude of a complex number- Arithmetic operations on complex numbers - Modulus-Amplitude (polar) form, Exponential form (Euler form) of a complex number.
11. **Hyperbolic functions:** Definition of hyperbolic and inverse hyperbolic trigonometric functions- and list formulae.

UNIT-III: Coordinate geometry

12 **Straight lines:** Various forms of a straight line - Angle between two lines, perpendicular distance from a point to the straight line, point of intersection of non-parallel lines and distance between parallel lines.
13. **Circle:** Locus of a point, Circle definition - Circle equation given (i) centre and radius, (ii) two ends of a diameter (iii) three non-collinear points of type (0,0), (a,0), (0, b) - General equation of a circle –its centre and radius.
14. **Conic sections**: Definition of a conic - Equation of a conic when focus, directrix and eccentricity are given - Properties of parabola in the standard form $y^2 = 4ax.$

UNIT-IV: Differential Calculus

15. **Concept of Limit:** Definition and Properties of Limits and Standard Limits - Continuity of a function at a point.
16. **Concept of derivative:** Definition (first principle)- different notations- Derivatives of standard algebraic, logarithmic, exponential, trigonometric, inverse trigonometric, hyperbolic and inverse hyperbolic functions - Derivatives of sum, difference, scalar multiplication, product, quotient of functions - Chain rule, derivatives of parametric functions, derivatives of implicit functions, logarithmic differentiation - Second order derivatives – Define maximum and minimum values of a function and find the maximum or minimum values for quadratic polynomial. Functions of several variables, first order partial derivatives.

UNIT-V: Integral Calculus

17. **Indefinite Integration:** Integration regarded as an anti-derivative – Indefinite integrals of standard functions. Properties of indefinite integrals. Integration by substitution or change of variable. Integrals of tan x, cot x, sec x and cosec x.
Evaluation of integrals which are of the following forms:

$i) \dfrac{1}{a^2+x^2}, \dfrac{1}{a^2-x^2}, \dfrac{1}{x^2-a^2}$

$ii) \dfrac{1}{\sqrt{a^2+x^2}}, \dfrac{1}{\sqrt{a^2-x^2}}, \dfrac{1}{\sqrt{x^2-a^2}}$

$iii) \sqrt{a^2+x^2}, \sqrt{a^2-x^2}, \sqrt{x^2-a^2}$

Integration by decomposition of the integrand into simple rational, algebraic functions - Integration by parts, Bernoulli's rule.
18. **Definite Integration:** Definite integral, fundamental theorem of integral calculus, properties of definite integrals, evaluation of simple definite integrals.

TEXTBOOK

Engineering Mathematics-I, a textbook for first year diploma courses, prepared & prescribed by SBTET, AP.

REFERENCE BOOKS

1. Shanti Narayan, A Textbook of matrices, S.Chand & Co.
2. Robert E. Moyer & Frank Ayers Jr., Schaum's Outline of Trigonometry, 4th Edition, Schaum's Series.
3. G.B.Thomas, R.L.Finney, Calculus and Analytic Geometry, Addison Wesley, 9th Edition, 1995.
4. Frank Ayers & Elliott Mendelson, Schaum's Outline of Calculus, Schaum's Series.
5. M.Vygodsky, Mathematical Handbook, Mir Publishers, Moscow.

SUGGESTED E-LEARNING REFERENCES

1. https://www.khanacademy.org/
2. https://www.wolframalpha.com/
3. https://onlinecourses.nptel.ac.in/
4. http://tutorial.math.lamar.edu/

UNIT TEST SYLLABUS

Unit Test	Learning Outcomes to be Covered
Unit Test – 1	From 1.1 to 2.11
Unit Test – 2	From 2.12 to 3.8
Unit Test – 3	From 4.1 to 5.14

Course Code	Course Title	No. of Periods per week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM103T	Engineering Physics	3	90	30	70	4

26CB103T- ENGINEERING PHYSICS

TIME SCHEDULE

S. No	Major Topics	No. of Periods	COs
1.	Units and Measurements	09	CO1
2.	Elements of Vectors	11	
3.	Mechanics	10	CO2
4.	Fundamentals of Astrodynamics	13	
5.	Energy and Thermal Physics	12	CO3
6.	Concepts of Acoustics	12	
7.	Electricity and Magnetism	13	CO4
8.	Modern Physics	10	
	Total	**90**	

COURSE OBJECTIVES

Upon completion of the course the student shall be able to

(1)	To understand the basic concepts of physics for various Engineering applications as required for industries.
(2)	To equip the students with the scientific advances in technology and make the student suitable for any industrial organization.

COURSE OUTCOMES

CO1	CM103T.1	Familiarize with various physical quantities, their SI units and errors in measurements; Understand the concepts of vectors for solving engineering problems.
CO2	CM103T.2	Solve problems in engineering using appropriate equations and formulae related to Mechanics; Understand the concepts of gravitation, planetary motion with reference to applications in satellites
CO3	CM103T.3	Familiarize with the knowledge of various forms of energy, thermal physics and concepts of acoustics in relevance to the societal requirements.
CO4	CM103T.4	Familiarize with the basic knowledge of electricity, magnetism and advances in Modern Physics such as photoelectric cell, optical fibers, superconductors and nanotechnology.

LEARNING OUTCOMES

1.0 Units and Measurements

1.1 Introduction to Units and Measurements

1.2 Define the terms: a) Physical quantity b) Fundamental physical quantities

and c) Derived physical quantities.

1.3 Explain the concept of units in measurement.

1.4 Define the term 'unit'.

1.5 Define fundamental units and derived units.

1.6 State the SI units of fundamental quantities along with their symbols.

1.7 State the common multiples and submultiples used in the SI system.

1.8 State the rules for writing SI units.

1.9 State the advantages of using SI units.

1.10 Differentiate between direct and indirect measurements.

1.11 Define accuracy and least count in the context of measurement.

1.12 Define error in measurement.

1.13 Define absolute, relative and percentage errors and state their respective

formulae.

1.14 Solve numerical problems on errors in measurements.

2.0 ELEMENTS OF VECTORS

2.1 Explain the concept of vectors.

2.2 Define scalar and vector quantities with relevant examples for each.

2.3 Represent a vector geometrically.

2.4 Define equal vectors, negative vector, unit vector, position vector,

co-initial vectors, co-planar vectors.

2.6 Resolve a given vector into its rectangular components.

2.7 State and explain the triangle law of addition of vectors.

2.8 State the parallelogram law of addition of vectors.

2.9 Derive the expressions for the magnitude and direction of the resultant

vector using the parallelogram law.

2.10 Illustrate applications of the parallelogram law of vectors using examples

(i) Bow and arrow (ii) working of a sling (iii) Flying of a bird.

2.11 Define dot product (scalar product) of two vectors.

2.12 Explain (i) work done (ii) power as examples of dot product.

2.13 Define cross product (vector product) of two vectors.

2.14 Explain (i) linear velocity (ii) torque as examples of cross product.

2.15 Solve numerical problems on (i) resolution of vectors (ii) the parallelogram

law of vectors iii) dot product.

3.0 MECHANICS

3.1 Define linear momentum; Mention its SI unit.

3.2 Define force. Mention its SI unit.

3.3 Define torque. Mention its SI unit.

3.4 Define concurrent forces, co-planar forces.

3.5 State and explain Lami's theorem.

3.6 State equations of motion of a body moving in a straight line with uniform

acceleration.

3.7 Define projectile. Give examples.

3.8 Derive the equation for the path of an oblique projectile.

3.9 Define periodic motion.

3.10 Define Ideal Simple pendulum.

3.11 Write formula for the time period of a simple pendulum.

3.12 Solve numerical problems on equations of motion and simple pendulum.

4.0 FUNDAMENTALS OF ASTRODYNAMICS

4.1 Define acceleration due to gravity (g); Mention its SI unit.

4.2 State and explain Newton's universal law of gravitation.

4.3 Define universal gravitational constant (G) and mention its value in SI unit.

4.4 Derive the relationship between acceleration due to gravity (g) and the universal gravitational constant (G).

4.5 State and explain Kepler's laws of planetary motion.

4.6 Define orbital velocity and state its formula.

4.7 Define escape velocity and state its formula.

4.8 Derive the relationship between escape velocity and orbital velocity.

4.9 Define the term 'satellite'.

4.10 Define natural and artificial satellites. Give examples for each.

4.11 Mention the applications of artificial satellites.

4.12 Solve numerical problems on (i) Newton's law of gravitation (ii) orbital velocity (iii) escape velocity.

5.0 ENERGY AND THERMAL PHYSICS

5.1 Define work done; Mention its SI unit.

5.2 Define power; Mention its SI unit.

5.3 Define energy; Mention its SI unit.

5.4 List various forms of energy.

5.5 Define potential energy; Give examples and derive its equation.

5.6 Define kinetic energy; Give examples and derive its equation.

5.7 Derive the relationship between kinetic energy and linear momentum.

5.8 State the law of conservation of energy; Give any two examples.

5.9 State Boyle's law; Write its equation.

5.10 State Charles's volume law; Write its equation.

5.11 State Charles's pressure law; Write its equation.

5.12 Define an Ideal gas.

5.13 Derive the ideal gas equation (PV = nRT).

5.14 Solve numerical problems on (i) Work done (ii) Potential energy (iii) Kinetic

energy (iv) Relation between K.E. and momentum (v) Gas laws

6.0 CONCEPTS OF ACOUSTICS

6.1 Define longitudinal waves. Give examples.

6.2 Define transverse waves. Give examples.

6.3 Define sound. Mention SI unit for intensity of sound.

6.4 Define musical sound.

6.5 Define noise.

6.6 Distinguish between musical sound and noise.

6.7 Define noise pollution.

6.8 Explain the sources of noise pollution.

6.9 Explain the effects of noise pollution.

6.10 Explain methods of minimizing noise pollution.

6.11 Explain the concept of echo.

6.12 Mention the applications of echo.

6.13 Define reverberation and reverberation time.

6.14 Write Sabine's formula and name the parameters in it.

6.15 Solve numerical problems on echo.

7.0 Electricity and Magnetism

7.1 State and explain Ohm's law.

7.2 Define electrical resistance; Mention its SI unit.

7.3 Define specific resistance (resistivity);Mention its SI unit.

7.4 State and explain Kirchhoff's Current Law.

7.5 State and explain Kirchhoff's Voltage Law.

7.6 Derive an expression for the balancing condition of Wheatstone's bridge with neat diagram.

7.7 Explain the concept of magnetic field.

7.8 Define uniform and non-uniform magnetic fields.

7.9 Define magnetic pole strength; Mention its SI unit.

7.10 Define magnetic moment; Mention its SI unit.

7.11 Define magnetic lines of force.

7.12 Write the properties of magnetic lines of force.

7.13 State Coulomb's inverse square law of magnetism. Write its equation.

7.14 Derive the expression for the moment of couple acting on a bar magnet placed in a uniform magnetic field.

7.15 Solve numerical problems on (i) Ohm's law (ii) Kirchhoff's first law (iii) Wheatstone bridge (iv) Coulomb's inverse square law of magnetism.

8.0 Modern Physics

8.1 State and explain photoelectric effect.

8.2 Write Einstein's photoelectric equation and name the terms in it.

8.3 Explain the working of a photoelectric cell.

8.4 List the applications of the photoelectric cell.

8.5 Define critical angle.

8.6 Explain the phenomenon of total internal reflection.

8.7 Define optical fiber; Explain the principle and working of an optical fiber.

8.8 List the applications of optical fiber.

8.9 Define Superconductor and superconductivity.

8.10 List the applications of superconductors.

8.11 Define Nanotechnology and Nano materials.

8.12 Write applications of Nano materials.

CO-PO MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	1	1	1		1
CO2	3	2	1	1	1		2
CO3	3	2	1	1	1		2
CO4	3	2	1	1	3		2

3 = strongly mapped, 2 = moderately mapped, 1 = slightly mapped

Note: The gaps in CO and PO mapping will be achieved by one or more appropriate activities from the following.

(i) Seminars (ii) Viva-voce (iii) Assignments (iv) Quiz competitions (v) Industrial visits (vi)Techfest (vii) Mini project (viii) Group discussions (ix) Virtual labs (x) Library visit for e-books

COURSE CONTENT

1. **Units and measurements:**

   Introduction – Physical quantity – Fundamental and Derived quantities –Unit-Fundamental and derived units - SI system of units –Multiples and Sub multiples – Rules for writing S.I. units-Advantages of SI units – Direct and indirect measurements – Accuracy and least count – Errors: Absolute, relative and percentage errors – Problems.

2. **Elements of Vectors:**

   Introduction of Scalars and Vectors – Representation of a vector –Types of vectors - Resolution of vector into rectangular components – Triangle law of vectors - Parallelogram law of vectors- examples- derivation of magnitude and direction of resultant vector- Dot product-Cross product - Problems.

3. **Mechanics**:

   Introduction to Mechanics – Momentum –force-torque. Concurrent and coplanar forces - Lami's theorem – equations of motion of a body moving in a straight line – projectile - path of projectile in oblique projection – periodic motion -Ideal simple pendulum- Time period of simple pendulum- Problems.

4. **Fundamentals of Astrodynamics:**

   Concept of acceleration due to gravity (g) -Newton's law of gravitation- Universal Gravitational constant G – Relation between g and G- Kepler's laws of planetary motion – Orbital velocity and escape velocity – Satellites: Natural and artificial - Applications of artificial satellites – Problems.

5. **Energy and thermal Physics**

   Work done, Power and Energy - forms of energy - Potential energy - Kinetic energy-Momentum- K.E and Momentum relation – Law of Conservation of energy- Boyle's law - Charle's volume law -Charle's pressure law- Ideal Gas equation- Problems.

6. **Concepts of Acoustics**

   Longitudinal wave- transverse wave- musical sound - noise - Noise pollution – Causes, effects, Methods of minimizing noise pollution- Echo- Reverberation - Reverberation time-Sabine 's formula - Problems.

7. **Electricity and Magnetism**

   Ohm's law- Resistance - Specific resistance - Kirchoff's laws - Wheatstone's bridge. Concept of magnetic field- magnetic pole strength – Magnetic Moment- magnetic lines of force - Coulomb's inverse square law of magnetism– Torque acting on a bar magnet- Problems.

8. **Modern Physics**

   Photoelectric effect – Einstein photo electric equation – photoelectric cell – Applications of photoelectric cell – critical angle, Total internal reflection- Optical Fiber - Principle – working-Applications of optical fibers - Superconductivity–applications – Nanotechnology – applications.

REFERENCES

1. Intermediate physics - Volume - I & 2
2. Telugu Academy (English version)
3. Unified physics Volume 1, 2, 3 and 4
4. Dr. S.L Guptha and Sanjeev Guptha
5. Concepts of Physics, Vol 1 & 2 — H.C. Verma
6. Text book of physics Volume I — Resnick & Holiday
7. Fundamentals of physics — Brijlal & Subramanyam
8. Text book of applied physics — Dhanpath Roy
9. NCERT Text Books of physics — Class XI & XII Standard
10. e-books/e-tools/websites/Learning Physics software

TABLE SHOWING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TESTS

Unit test - 1	From 1.1 to 3.12
Unit test - 2	From 4.1 to 6.15
Unit test - 3	From 7.1 to 8.12

CURRICULUM –2026

C – 26

ENGINEERING CHEMISTRY

AND

ENVIRONMENTAL STUDIES

[COMMON-104]

STATE BOARD OF TECHNICAL
EDUCATION & TRAINING
ANDHRA PRADESH

ENGINEERING CHEMISTRY AND ENVIRONMENTAL STUDIES

Course Code	Course Title	No. of Periods per Week	Total No. of Periods	Marks for FA	Marks for SA	Credits
26CM104T	Engineering Chemistry and Environmental Studies	3	90	30	70	4

TIME SCHEDULE

Chapter No.	Unit Title/Chapter	No. of Periods	Weightage	No. of Short Questions (3 marks)	No. of Essay Questions (8 marks)	COs Mapped
1	Basic Concepts of Chemistry	14	18	2	1.5	CO1
2	Solutions, Acids and Bases	14	15	1	1.5	CO1
3	Electrochemistry	12	11	1	1	CO2
4	Corrosion	8	11	1	1	CO2
5	Water Treatment	8	11	1	1	CO3
6	Polymers and Engineering Materials	10	6	2	0	CO4
7	***Fuels and Alternative Energy Sources***	6	3	1	0	CO4
8	Environmental Studies	18	19	1	2	CO5
TOTAL		**90**	**94**	**10**	**8**	**CO1,CO2, CO3,CO4, CO5**

COURSE OBJECTIVES

Upon completion of course the student shall be able to	
(i)	To develop a fundamental understanding of core chemical principles and their relevance to a wide range of engineering applications.
(ii)	To explore and analyze natural and anthropogenic environmental challenges through an interdisciplinary lens, incorporating physical, chemical and socio - cultural perspectives.
(iii)	To reinforce theoretical concepts by conducting relevant experiments exercises

COURSE OUTCOMES

CO1	104.1&104.2	Explain the basics of atomic structure, chemical bonding, oxidation-reduction, mole concept, concentration expressing methods of solutions, acids-bases, pH and buffer solutions.
CO2	104.3 & 104.4	Explain electrolysis, Galvanic cell, batteries and corrosion.
CO3	104.5	Explain the chemistry involved in the treatment of hardness in water.
CO4	104.6 & 104.7	Explain the preparation and applications of polymers, and understand the composition and uses of alloys, nanomaterials and green fuels.
CO5	104.8	Explain environmental concepts, pollution types, global issues, green chemistry principles and sustainable development goals.

LEARNING OUTCOMES

1.0 Basic Concepts of Chemistry

1.1 Explain the charge, mass of fundamental particles of an atom (electron, proton and neutron).

1.2 Understand the concept of Atomic number and Mass number.

1.3 Calculate the number of electrons, number of protons and number of neutrons in atoms, if Atomic number and Mass number are given.

1.4 Explain the Postulates of Bohr's atomic theory and its limitations.

1.5 Explain the values and significance of four Quantum numbers.

1.6 Define Orbital of an atom and draw the shapes of s, p orbitals.

1.7 Distinguish between orbit and orbital.

1.8 Explain (i). Aufbau principle (ii). Hund's rule and (iii). Pauli's exclusion principle.

1.9 Write the Electronic configuration of elements up to Atomic number 20.

1.10 Explain the significance of chemical bonding.

1.11 Understand the concept of Octet rule.

1.12 Define Ionic bond and explain it in the formation of NaCl.

1.13 Define Covalent bond and explain it in the formation of H_2, O_2 & N_2 molecules (Lewis Dot Method).

1.14 List out the Properties of Ionic compounds and Covalent compounds and distinguish between their properties.

1.15 Understand the electronic concept of oxidation, reduction and redox reactions

2.0 Solutions, Acids and Bases

2.1 Define the terms: (i). Solution (ii). Solute and (iii). Solvent with examples.

2.2 Classify solutions based on physical state of solvent with examples.

2.3 Define the terms: (i). Atomic weight, (ii). Molecular weight, and (iii). Equivalent

weight.

2.4 Calculate Molecular weight and Equivalent weight of the given Acids (HCl, H_2SO_4, H_3PO_4),Bases (NaOH, $Ca(OH)_2$, $Al(OH)_3$ and Salts (NaCl, Na_2CO_3, $AlCl_3$).

2.5 Define Mole and solve numerical problems on Mole concept.

2.6 Define Molarity, Normality and solve numerical problems on Molarity and Normality.

(a). Calculate the Molarity & Normality, if Weight of solute and Volume of solution are given.

(b). Calculate the weight of solute, if Molarity or Normality with volume of solution are given.

2.7 Explain Arrhenius theory of Acids and Bases and give its limitations.

2.8 Define pH and mention its Significance.

2.9 Define buffer solution and classify buffer solutions with examples. Give its applications.

3.0 Electrochemistry

3.1 Define the terms (i). Conductor (ii). Semiconductor (iii). Insulator. (iv). Electrolyte (Strong and Weak) and (v). Non-electrolyte. Give two examples for each.

3.2 Define Electrolysis and Explain electrolysis by taking an example of molten NaCl.

3.3 State the applications of electrolysis.

3.4 Understand Electrode potential and Standard reduction potential (SRP).

3.5 Define electrochemical series and state its significance.

3.6 Define Galvanic cell. Explain the construction and working of Galvanic cell.

3.7 Distinguish between electrolytic cell and galvanic cell.

3.8 Define battery and list the types of batteries with examples.

3.9 Explain the construction, working and applications of (i). Dry cell (Leclanché cell) and (ii). Lithium-ion battery.

4.0 Corrosion

4.1 Define the term corrosion.

4.2 State the factors which influencing the rate of corrosion.

4.3 Describe the formation of (a). Composition cell (b). Stress cell and (c). Concentration cell during corrosion.

4.4 Define rusting of iron and explain the mechanism of rusting of iron.

4.5 Explain the methods of prevention of corrosion by:

(a). Protective Coatings (through flow chart with examples) and

(b). Cathodic Protection Methods. ((i). Sacrificial Anode Process and (ii). Impressed Voltage Process)

5.0 Water Treatment

5.1 Define soft water and hard water.

5.2 Define hardness of water and classify its types.

5.3 List out the salts that causing hardness of water (with Formulae).

5.4 State the disadvantages of using hard water in industries.

5.5 Define Degree of hardness and units of hardness (mg/L and ppm).

5.6 Explain the method of softening of hard water by Ion exchange method (By indicative reactions).

5.7 Explain the concept of Reverse Osmosis in removing hardness of water.

5.8 List out the applications and advantages of reverse osmosis technique.

5.9 List out the essential qualities of drinking water/potable water.

5.10 Explain Municipal treatment of water for drinking purpose (only flow chart).

6.0 Polymers and Engineering Materials.

6.1 Explain monomers, polymers and the concept of polymerization.

6.2 Describe the methods of polymerization (a). Addition Polymerization of Polythene and (b). Condensation Polymerization of Bakelite (Only flow chart).

6.3 Define plastic. Write the monomers and uses of plastics:
(i). PVC and (ii) Nylon (6,6).

6.4 Define Biodegradable polymers. State applications of (i). PHBV and (ii). PBAT.

6.5 Define an alloy. Write the composition and applications of the following alloys:
(i). Stainless Steel and (ii). Nitinol.

6.6 Define Nano Materials and State applications of
(i). Graphene and (ii). Nanotubes.

7.0 Fuels and Alternative Energy Sources

7.1 Define the term fuel.

7.2 Classification of fuels as Natural fuels and Synthetic fuels.

7.3 Write the composition and uses of the following:
(i). LPG (ii). CNG and (iii). Power alcohol.

7.4 State the Renewable and Non- renewable energy sources with examples.

7.5 Define Green fuel. State the advantages and disadvantages of hydrogen as a green fuel.

8.0 Environmental Studies

8.1 Importance of environmental studies.

8.2 Define the following terms:
(i). Pollution, (ii). Pollutant, (iii). Sink, (iv). Receptor, (v). Particulate Matter, (vi). Dissolved Oxygen (DO) and (vii). Threshold Limit Value (TLV).

8.3 State the uses of forest resources.

8.4 Define deforestation. Explain the causes, effects and controlling methods of deforestation.

8.5 Define Air pollution. Explain the causes, effects and controlling methods of Air pollution.

8.6 Explain the global impacts of Air pollution: (i). Global Warming, (ii). Ozone Layer Depletion and (iii). Acid Rain.

8.7 Define Water pollution. Explain the causes, effects and controlling methods of Water pollution.

8.8 Define e – pollution. State the sources of e – pollution. Explain its health effects and its management.

8.9 Define Green Chemistry. List the Green Chemistry Principles.

8.10 Define Sustainable Development and List the Sustainable Development Goals.

CO-PO/PSO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1	3	-	-	-	-	-	-	-	-	-
CO2	3	1	1	1	1	-	1	-	-	-
CO3	3	1	1	1	1	-	1	-	-	-
CO4	3	1	1	-	1	-	1	-	-	-
CO5	3	1	-	-	1	1	1	-	-	-
Average	**3**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**0**	**0**

3 = Strongly mapped 2 = Moderately mapped 1 = Slightly mapped

Note: The gaps in CO and PO mapping will be achieved by one or more appropriate activities from the following:

(i). Seminars (ii). Tutorials (iii). Guest Lectures (iv). Assignments (v). Quiz Competitions,
(vi). Industrial Visit (vii). Tech Fest (viii). Mini Project (ix). Group Discussions
(x). Virtual Classes and (xi). Library Visits etc.

COURSE CONTENT

1.Basic Concepts of Chemistry

Atomic Structure:

Introduction - Fundamental particles – their mass and charge – Atomic number and Mass number - definition with examples – calculation of electrons, protons and neutrons in atoms – Bohr's atomic theory and limitations - Quantum numbers – Orbital concept, shapes of s, p orbitals – Distinguish between orbit and orbital - Aufbau principle - Hund's rule - Pauli's exclusion Principle - Electronic configuration of elements (Atomic number(Z) from 1 to 20).

Chemical Bonding:

Introduction – Octet rule - Types of chemical bonds – Ionic bond (NaCl) and Covalent bond (H_2, O_2 & N_2 molecules) as examples – Properties of Ionic and Covalent compounds. Electronic concept of oxidation, reduction and redox reactions.

2.Solutions, Acids and Bases

Solutions:

Introduction – Idea of solute, solvent and solution - Types of solutions based on physical state of solvent – Atomic weight – Molecular weight, Equivalent Weight (Acids, Bases and Salts) - Mole concept – Numerical problems on Mole concept - Methods of expressing concentration of a solution – Molarity - Normality – Numerical problems on Molarity and Normality.

Acids and Bases:

Introduction - Arrhenius theory of acids and bases – pH Scale – its significance – Buffer solution – Definition – Types of buffer solutions with examples – its applications.

3.Electrochemistry

Introduction - Conductors, Semiconductors, Insulators with examples - Electrolytes (Strong and Weak) and Non-electrolytes – Definition – Examples – Electrolysis – Definition – Electrolysis of molten NaCl – Applications of electrolysis – Electrode potential - Standard reduction potential – Definition – Electrochemical series – Significance – Construction and working of Galvanic cell – Differences between Electrolytic cell and Galvanic cell - Batteries - Types of batteries – Definition and examples – construction, working and applications of: (i). Dry Cell (Leclanché Cell) and (ii). Lithium-ion battery.

4.Corrosion

Introduction – Definition - Factors influencing the rate of corrosion – Composition cell, Stress cell and Concentration cell during corrosion – Rusting of iron and its mechanism – Prevention of corrosion - Protective Coating methods (flow chart with examples) - Cathodic Protection methods.

5.Water Treatment

Introduction – Soft and Hard water – Hardness of water – Types of hardness – salts responsible for hardness - Degree of hardness – Methods of expressing hardness (mg/L and ppm) – Disadvantages of using hard water in industries - Softening of hard water by Ion exchange method – Concept of Reverse Osmosis process – Applications and Advantages of Reverse Osmosis - Essential qualities of drinking water/potable water – Municipal treatment of water for drinking purpose (only flow chart).

6.Polymers and Engineering Materials

Polymers:

Introduction- Monomers - Polymers - Polymerization – Types of Polymerization – Addition polymerization (Polythene) and Condensation polymerization (only flow chart of Bakelite) - Plastics – monomers and uses of PVC and Nylon (6,6) - Biodegradable Polymers: (i). PHBV and (ii). PBAT (Composition and Uses).

Engineering Materials:

Alloys - Definition - Composition and applications of (i). Stainless Steel and (ii). Nitinol

Nano Materials – Definition - Applications of (i) Graphene and (ii). Nanotubes.

7.Fuels and Alternative Energy Sources

Introduction – Definition - Classification of fuels – Composition and uses of (i). LPG (ii). CNG and (iii). Power alcohol - Renewable and Non-renewable energy sources – Advantages and disadvantages of Hydrogen as a green fuel.

8.Environmental Studies

Introduction - Importance of environmental studies – Important terms related to environment – Pollution, Pollutant, Sink, Receptor, Particulate Matter, Dissolved Oxygen (DO), Threshold Limit Value (TLV) - Uses of forest resources – Deforestation - Definition – causes, effects, controlling methods – Air pollution – Definition, causes, effects, controlling methods - Global impacts of Air pollution – Global warming, Ozone layer depletion, Acid rain – Water pollution – Definition, causes, effects, controlling methods – e - pollution, Definition, sources, effects, management - Green Chemistry – Definition – Principles of Green Chemistry – Sustainable Development – Definition – Goals.

REFERENCE BOOKS

1. **Jain & Jain-Engineering Chemistry**
2. **O.P. Agarwal, Hi-Tech-Engineering Chemistry**
3. **B. K. Sharma-Engineering Chemistry**
4. **A. K. De-Engineering Chemistry**

5. **Mahua Basu & S. Xavier-Fundamentals of Environmental Studies**
6. **Anubha Kaushik & C.P Kaushik-Environmental Studies**

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TEST-I, UNIT TEST-II & UNIT TEST-III

Unit Test	**Learning outcomes to be covered**
Unit Test – 1	From 1.1 to 2.9
Unit Test – 2	From 3.1 to 5.10
Unit Test – 3	From 6.1 to 8.10

PYTHON PROGRAMMING

Course code	Course Title	No. of Periods/Week	Total No. of periods	Marks for FA	Marks for SA
26CM105T	Python Programming	6	180	30	70

Time Schedule

Chapter No.	Chapter/Unit Title	No. of Periods	Marks	No. of Short Answer Questions	No. of Essay Type Questions	CO's Mapped
1.	Introduction to Python Programming	40	14	02	01	CO1
2.	Control Flow and Loop statements	25	14	02	01	CO2
3.	Functions and Arrays	40	22	02	02	CO3
4.	Data Structures	35	22	02	02	CO4
5.	Object Oriented Programming basics, Python Libraries and File Handling	40	22	02	02	CO5
	Total	**180**	94	10	08	

Course Objectives	i)To know the fundamentals of Python programming and to understand syntax, scripting and program execution in 'Python' ii)To develop various python programs using control structures, functions & arrays iii) To develop programs using data structures iv) To Implement OOPs basic concept and Python Libraries v) To implement file handling concepts

Course Outcomes	CO1	CM105T.1	Explain Basic concepts of Python programming as well as scripting, debugging and execution.
	CO2	CM105T.2	Develop Python programs using Control statements & Loops.
	CO3	CM105T.3	Develop Python programs using Functions and arrays.
	CO4	CM105T.4	Develop Python programs using Data structures.

	CO5	CM105T.5	Develop Python application programs using OOP basic concepts, Python libraries and Files,

Learning Outcomes:

1.0 **Introduction to python programming**

1.1 Recite history of Python.
1.2 List Python features
1.3 List and Explain Applications of Python
1.4 Explain Python Integrated Development and Learning Environment (IDLE)
1.5 Explain process of Running Python Scripts.
1.6 Explain Identifiers, Keywords, Indentation & Variables
1.7 List and explain various data types
1.8 Explain declaration & initialization of variables.
1.9 Explain Input and Output statements.
1.10 Explain formatted input output.
1.11 State the usage of comments
1.12 List and Explain various Operators.
1.13 Explain Boolean values.
1.14 Explain Operator precedence rules.
1.15 State the purpose of modules.
1.16 Define functions.
1.17 List types of functions
1.18 List and Explain Built-in Functions.
1.19 Explain the Steps in Developing a simple python program and execution.

2.0 **Control Flow and Loop Statements**

2.1 List and Explain various Control Flow constructs.
- 2.1.1 If
- 2.1.2 If-Else
- 2.1.3 if-elif-else

2.2 List and Explain various Loop Statements.
- 2.2.1 for Loop
- 2.2.2 while loop

2.3 Use break, continue & pass statements.
2.4 Implement the python programs using control structures

3.0 **Functions and Arrays**

3.1 Give introduction to functions
3.2 Use Function Arguments: Default arguments, Keyword arguments, Variable Length arguments, Anonymous Functions & Return Statement.
3.3 List and explain Scope of variables
3.4 Explain creation of modules.
3.5 Explain importing of modules.
3.6 Explain Python Variables: Namespace and scoping
3.7 Explain Python Packages
3.8 List and Explain Strings: String slices, immutability
3.9 List and Explain String functions and methods.
3.10 Explain about String module.
3.11 Explain about Python Arrays.
3.12 Explain accessing of elements in an Array.
3.13 Explain Array methods.

4.0 Data Structures

4.1 Explain Python Lists.
4.2 Describe Basic List Operations.
4.3 Explain List Slices.

4.4 Explain List methods.
4.5 Explain List traversing.
4.6 Explain mutability.
4.7 Explain aliasing.
4.8 Explain Cloning of lists.
4.9 Explain List comprehension.
4.10 Tuples.
4.10.1 Explain Tuple assignment.
4.10.2 Explain Tuple as return value.
4.10.3 Explain Tuple Comprehension
4.11 Dictionaries
4.11.1 Explain creation of dictionary.
4.11.2 Explain Dictionary operations and methods.
4.11.3 Explain Dictionary Comprehension.
4.12 Explain Sets.

5.0 Object oriented programming, Python Libraries & File Handling

5.1 Create Classes
5.2 Create Objects
5.3 Explain Data Hiding
5.4 Explain Data Abstraction
5.5 Explain Inheritance.
5.6 Explain method overloading and overriding.
5.7 Give introduction to python library.
5.8 Import modules in a python program
5.9 Use Python standard library's functions and modules
5.9.1 Turtle, Random, Datetime, CSV, Numpy & Pandas
5.10 Create Python library
5.11 Give introduction to Files
5.12 Explain opening and closing Files
5.13 Explain reading and writing Files
5.14 Use Standard Input, Output and error streams
5.15 Explain exception handling
5.15.1 Define Error and Exception
5.15.2 State the difference between Error and Exception
5.15.3 Use user-defined exceptions.
5.15.4 Use try except block.
5.15.5 Explain raising exceptions.

CO-PO/PSO Matrix:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM105T.1	3	1	2	1	1	1		2	1	
CM105T.2	3	2	2	1	1	1	1	2	2	2
CM105T.3	3	2	2	1	1	1		2	2	2
CM105T.4	3	1	2	1	1	1	1	2	2	2
CM105T.5	3	1	2	1	2	3	2	2	2	2
Average	3	1.75	2	1	1.2	1.8	1.3	2	1.8	1.8

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

COURSE CONTENT

UNIT – I: Introduction

Introduction to Python, installation, Applications, IDLE, Data

types, variables, Input and Output statements, Operators, expressions, statements, precedence of operators, comments, modules, functions, types of functions, built-in functions, flow of python program execution.

UNIT – II: Control Flow and Loop Statements:

Control Flow- If, if-else, if-elif-else, for, while, break, continue, pass statements.

sample python programs using control flow and Loop statements.

UNIT – III: Functions and Arrays:

Introduction to Functions: Defining, Calling Functions, Passing Arguments, Types of function arguments -Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Function Returning multiple values, Scope of the Variables in a Function - Global and Local Variables.

Modules: Creating modules, import statement, name spacing, Python Packages.

Strings: string slices, immutability, string functions and methods, string module.

Python arrays: Access the Elements of an Array, array methods.

UNIT – IV: Data Structures

Lists: list operations, list slices, list methods, list traversing, mutability, aliasing, cloning lists, list comprehension.

Tuples: tuple assignment, tuple as return value, tuple comprehension.

Dictionaries: operations and methods, comprehension.

sets.

UNIT – V: Object Oriented Programming , Python Libraries & File Handling

Object Oriented Programming in Python: Basics of Classes, Objects, Methods, Constructor, Data hiding, Abstraction, Inheritance, overriding.

Python Libraries: Introduction, importing modules in python program, use python standard library functions and modules - Python libraries - Turtle, Random, Datetime, CSV, Numpy & Pandas.

File Handling: Open Files, File Processing and Closing a File.

Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions

REFERENCE BOOKS:

1. Python Programing by K. Nageswara Rao, Shaikh Akbar -Scitech Publications (India) Pvt. Ltd.
2. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
3. Learning Python, Mark Lutz, Orielly
4. Think Python, Allen Downey, Green Tea Press
5. Core Python Programming, W.Chun, Pearson.
6. Introduction to Python, Kenneth A. Lambert, Cengage

Table specifying the scope of syllabus to be covered for unit tests

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 2.1
Unit test-2	From 2.2 to 3.13
Unit test-3	From 4.1 to 5.15

CODING FUNDAMENTALS

Course code	Course Title	No. of Periods/Week	Total No. of periods	Marks for FA	Marks for SA
26CM106A	CODING FUNDAMENTALS	2	60	NIL	NIL

TIME SCHEDULE

S.No.	Chapter/Unit Title	No. of Periods	CO's Mapped
1.	Fundamentals of Computers	10	CO1
2.	Programming Methodology	15	CO2
.	Introduction to coding	10	CO3
4.	Fun with functions	15	CO4
5.	Understanding arrays and collections	10	CO5
	Total Periods	60	

Course Objectives

Course Objectives	i) To know the fundamentals of Computers ii) To familiarize with programming methodologies like algorithms and flowcharts iii) To understand coding basics iv) To familiarize with functions v) To familiarize with arrays and collections

Course Outcomes:

At the end of the course the student able to:		
CO1	CM106A.1	Explain computer fundamentals
CO2	CM106A.2	Explain various flowchart, algorithm methods
CO3	CM106A.3	Explain the concept of coding and basics of coding
CO4	CM106A.4	Explain the concept of functions, events and event handlers
CO5	CM106A.5	Explain the arrays and collections

Learning Outcomes:

1.0 Fundamentals of Digital Computer

1.1. Define various terms related to computers – Computer, Hardware, Software, Firmware, High Level Language, Low Level Language
1.2. Draw and explain block diagram of a computer in detail
1.3. Describe the current family of CPUs used in Computers.
1.4. State the use of storage devices used in a computer.
1.5. List the two types of memory used in a computer.

1.6. State the importance of cache memory.
1.7. Explain the generations of computers.
1.8. Classification of computers - based on a) size, b) processor.
1.9. State the importance of binary number system for use in Digital Computers

2.0 Implement Programming Methodology.

2.1. State the different steps involved in problem solving.
2.2. Define an algorithm.
2.3. List four characteristics of algorithm.
2.4. Define flowchart
2.5. Define a program
2.6. Differentiate between program and algorithm.
2.7. State the steps involved in algorithm development.
2.8. Differentiate between algorithm and flowchart.
2.9. Develop algorithms for simple problems.
2.10. Draw the various symbols used in flowcharts.
2.11. Draw flowcharts for simple problems.

3.0 Introduction to coding

3.1. Define coding.
3.2. Define Pseudocode
3.3. Explain the Process of writing code
3.4. Define variables.
3.5. Explain naming of variables
3.6. List Data types.
3.7. Explain operations on variables.
3.8. Explain AND, OR and NOT logical operators
3.9. State Combining of logical operators.
3.10. List different Relational operators
3.11. List Different control statements
3.12. Explain Nested control statements
3.13. Define loops
3.14. Explain Increment loops
3.15. List Different types of loops
3.16. State Entry criteria for a loop
3.17. State Exit criteria for a loop
3.18. Explain Break statement
3.19. Explain Continue statement
3.20 List Examples of sequential execution, selection and iteration
3.21 Define bug

4.0 Fun with functions

4.1 Define functions.
4.2 Examples of functions
4.3 List Advantages using functions
4.4 State different function parameters.
4.5 State about returning a value in a function.
4.6 Define an event.
4.7 List different event handlers.

5.0 Understanding arrays and collections

5.1. Explain collections.
5.2. Define arrays.
5.3 List various types of arrays with examples.
5.4 Explain the process of iterating over collections.
5.5 Explain modifying collections

CO-PO/PSO Matrix:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM106A.1	3							3	1	1
CM106A.2	1	2	2	1		3		1	3	1
CM106A.3	3	1		1		1	1	3	1	1
CM106A.4	3		2	2	1		1	2	1	2
CM106A.5	3		2		1	1	1	2	1	2
Average	2.7	1.5	2	1.25	1.3	1.7	1.25	2.2	1.3	1.3

3=Strongly mapped , 2=moderately mapped, 1=slightly mapped

COURSE CONTENT

1.0 Fundamentals of Digital Computer

Computer, Hardware, Software, Firmware, High Level Language, Low Level Language - Block diagram of a digital computer, Clock speed and word length, Functional blocks of a CPU: ALU and Control unit, classification of computer memory, generations & classification of computers, binary number system.

2.0 Programming Methodology.

Steps involved in problem solving – Define an algorithm , Program - Characteristics of algorithm - Differentiate between program and algorithm- Steps involved in algorithm development - Differentiate algorithm and flowchart - Algorithms for simple problems - Symbols used in flowcharts -Flowcharts for simple problems.

3.0 Introduction to coding

Need for coding – correlate coding with real life examples - pseudo code – process of writing pseudo code – Define Variables - naming of variables-Different data types in variables- operations on variables –Different types of logical operators and their functions i.e AND , OR and NOT – combining logical AND , OR and NOT operators - different types of relational operators -Different types of control statements if, if-else - Nested control statements - loops in coding-increment loops-decrement loops – entry criteria-exit criteria-break statement-continue statement-sequential execution, selection and iteration-Define bug

4.0 Fun with functions

Functions-examples-user defined and built-in functions- Different function parameters-returning a value in a function - Events- Event Handlers

5.0 **Understanding arrays and collections**

Collections-Arrays – examples for arrays and collections-Process of iterating over collections-Modifying collections.

REFERENCE BOOKS

1. Information Technology - Curtin.
2. Computer Science Theory & Application - E. Balaguruswamy, B. Sushila
3. Introduction to Computers (Special Indian Edition) - Peter Norton
4. How to solve it by computer-R.G.Dromey-Pearson Education
5.introduction to coding-class VI,VII,VIII,IX,X by cbse academic https://cbseacademic.nic.in/web_material/codeingDS/classVI_Coding_Student_Handbook.pdf

PYTHON PROGRAMMING LAB

Course code	Course Title	No. of Periods/Week	Total No. of periods	Marks for FA	Marks for SA
26CM107L	Python Programming Lab	6	180	40	60

TIME SCHEDULE:

Chapter No.	Chapter/Unit Title	No. of Periods	CO's Mapped
1.	Exercises on basics, expressions and operators.	30	CO1
2.	Exercises on control structures,	35	CO2
3.	Exercises on functions and Arrays	35	CO3
4.	Exercises on Data structures	40	CO4
5.	Exercise on Object orientation, python libraries and file handling	40	CO5
	Total Periods	180	

COURSE OUTCOMES

CO1	CM107L .1	Execute Simple python programs (basics, expressions and operators)
CO2	CM107L.2	Execute Python programs using control structures
CO3	CM107L.3	Execute python programming using Functions, Arrays
CO4	CM107L.4	Develop Python programs using Data structures
CO5	CM107L.5	Develop Python programs using OOP Concepts, Python libraries and File handling

LEARNING OUTCOMES

The student can able to:

1. Write and execute simple python Program.
2. Write a program to demonstrate various data types in Python
3. Write /execute simple 'Python' program: Develop minimum 2 programs using different data types (numbers, string, tuple, list, and dictionary).

4. Write a program to perform different Arithmetic Operations on numbers in Python
5. Write /execute simple 'Python' program: Develop minimum 2 programs using Arithmetic Operators, exhibiting data type conversion.
6. Write a Python script that prints prime numbers less than 20.
7. (i)Write simple programs to convert U.S. dollars to Indian rupees.
   (ii) Write simple programs to convert bits to Megabytes, Gigabytes and Terabytes.
8. Write simple programs to calculate the area and perimeter of the square, and the volume &perimeter of the cone.
9. Write program to: (i) determine whether a given number is odd or even.
   (ii) Find the greatest of the three numbers using conditional operators.
10. Write a program to: i) Find factorial of a given number. ii) Generate multiplication table up to 10 for numbers 1 to 5.
11. Write a python program to find factorial of a number using Recursion
12. Write a program to To print Factors of a given Number.
13. Write a program to: i) Find factorial of a given number. ii) Generate multiplication table up to 10 for numbers 1 to 5 using functions.
14. Write a program to: i) Find factorial of a given number using recursion. ii) Generate Fibonacci sequence up to 100 using recursion.
15. Write a python program to define a module to find Fibonacci Numbers and import the module to another program
16. Write a python program to define a module and import a specific function in that module to another program
17. Write a program to create, concatenate and print a string and accessing sub-string from a given string
18. Write a Python class to reverse a string word by word.
19. Write a program to Create a list, add element to list, delete element from the lists.
20. Write a program to Sort the list, reverse the list and counting elements in a list.
21. Write a program to demonstrate working with tuples in python.
22. Write a program to Create dictionary, add element to dictionary, delete element from the dictionary.
23. Write a program to To calculate average, mean, median, and standard deviation of numbers in a list.
24. Write a program to: i) To create simple file and write "Hello World" in it.
    ii) To open a file in write mode and append Hello world at the end of a file.
25. Write a program to: i) To open a file in read mode and write its contents to another file but replace every occurrence of character 'h' ii) To open a file in

read mode and print the number of occurrences of a character 'a'.

26. Write a program that inputs a text file. The program should print all of the unique words in the file in an alphabetical order
27. Write a Program to Add two complex number using classes and objects.
28. Write a Program to Subtract two complex number using classes and objects.
29. Write a pandas program to load the CSV into a dataframe and perform arithmetic operations on the data loaded.
30. Write a program to create a 5 * 5 Numpy array with random integers between 1 and 100.

CO-PO/PSO MATRIX

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM107L.1	2	2	2	1	2			3		2
CM107L.2	2	3	2					2		2
CM107L.3	3	3	2	3		2	2	2		
CM107L.4	2	2	2		2	3	1	2	3	
CM107L.5	3	3	2		2	2	2	2	2	
Average	2.3	2.3	2	2.3	2	2.3	2	2	2.5	2

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

S.No.	Name of the Experiment	Objectives	Key Competencies
1.	Write and execute simple python Program.	Write a simple python program to print Hello World! and debug and execute	1. Know the usage of Python IDLE 2. Edit and save the program 3. Check for the syntax errors and clear the errors 4. Run the program and check for the output.
2.	Write a program to demonstrate different number data types in Python	Write a simple python program using integer, float, string and debug and execute	1. Know the usage of Python IDLE 2. Edit and save the program **3.** Check for the syntax errors and clear the errors **4.** Run the program and check for the output.
3.	Write /execute simple 'Python' program: Develop minimum 2 programs using different data types (numbers, string, tuple, list, dictionary).	Write a Python program to identify different data types.	1. Identify different data types 2. Write basic python program using data types 3. Evaluate arithmetic expression 4. Run the program

			5. Rectify the syntactical errors 6. Execute the program and check the output for its correctness
4.	Write a program to perform different Arithmetic Operations on numbers in Python	Write a program to perform different Arithmetic Operations such as addition, subtraction, multiplication and division	1. Identify different arithmetic operators 2. Write basic python program using arithmetic operations 3. Evaluate arithmetic expression 4. Run the program 5. Rectify the syntactical errors **6.** Execute the program & Check the output for its correctness
5.	Write /execute simple 'Python' program: Develop minimum 2 programs using Arithmetic Operators, exhibiting data type conversion.	Write a Python program to identify arithmetic operators and data type conversion	1. Identify different arithmetic operators 2. Build arithmetic expressions 3. Identify the priorities of operators 4. Evaluate arithmetic expression 5. Run the program 6. Rectify the syntactical errors 7. Execute the program and Check the output for its correctness
6.	Write a Python script that prints prime numbers less than 20.	Write a Python script that prints prime numbers less than 20.	1. Identify 20 prime numbers 2. Edit and save the program **3.** Check for the syntax errors and clear the errors **4.** Run the program and check for the output.
7.	(i) Write simple programs to convert U.S. dollars to Indian rupees. ii) Write simple programs to convert bits to Megabytes, Gigabytes and Terabytes.	Write a Python program to identify arithmetic operators and data type conversion	• Identify different arithmetic operators • Build arithmetic expressions • Identify the priorities of operators • Evaluate arithmetic expression • Run the program • Rectify the syntactical errors • Execute the program Check the output for its correctness
8	Write simple programs to calculate the area and perimeter of the square, and the volume & perimeter of the cone.	Write a Python program to identify arithmetic operator data type conversion	1. Identify different arithmetic operators 2. Build arithmetic expressions 3. Identify the priorities of operators 4. Evaluate arithmetic

			expression 5. Run the program 6. Rectify the syntactical errors 7. Execute the program Check the output for its correctness
9	Write program to: (i) Determine whether a given number is odd or even. (ii) Find the greatest of the three numbers using conditional operators.	Write a Python program to identify conditional statements in Python.	1. Build a relational expression 2. Use the if statement for decision making 3. Rectify the syntax errors 4. Check the output for correctness
10	Write a program to: i) Find factorial of a given number. ii) Generate multiplication table up to 10 for numbers 1 to 5.	Write a Python program to identify loops statements in Python.	1. Build the termination condition for looping 2. Use while statement with correct syntax 3. Check whether correct number of iterations are performed by the while loop 4. Rectify the syntax errors 5. Debug logical errors
11	Write a python program to find factorial of a number using Recursion	Write a python program for the factorial for a given number using recursion	6. Build the application using recursion. 7. Build the terminating condition for recursion. 8. Rectify the syntax errors 9. Debug logical errors
12	Write a program to: To print Factors of a given Number.	Write a Python program to make use of function.	10. Build the termination condition for looping 11. Use while statement with correct syntax 12. Check whether correct number of iterations are performed by the while loop 13. Rectify the syntax errors 14. Debug logical errors
13	Write a program to: i) Find factorial of a given number. ii) Generate multiplication table up to 10 for numbers 1 to 5 using functions	Write a Python program to using functions	15. Build the termination condition for looping 16. Use while statement with correct syntax 17. Check whether correct number of iterations are performed by the while loop 18. Rectify the syntax errors 19. Debug logical errors
14	Write a program to: i) Find factorial of a given number using recursion. ii) Generate Fibonacci sequence up to 100 using recursion.	Write a Python program to make use of recursion.	20. Build the application using recursion. 21. Build the terminating condition for recursion. 22. Rectify the syntax errors 23. Debug logical errors
15.	Write a python program to define a module to find Fibonacci Numbers and import the module to another program	Write a Python program using module and importing	24. Build the application using module. 25. Develop the logic for

		the module	Fibonacci series. 26. Import the module 27. Rectify the syntax errors 28. Debug logical errors
16.	Write a python program to define a module and import a specific function in that module to another program	Write a Python program using module and importing the module	29. Build the application using module. 30. Develop the logic for Fibonacci series. 31. Import the module 32. Rectify the syntax errors 33. Debug logical errors
17.	Write a program to create, concatenate and print a string and accessing sub-string from a given string	Write a program to create, concatenate and print a string and accessing sub-string from a given string.	34. Build the application 35. Develop the logic for string concatenation and substring 36. Rectify the syntax errors 37. Debug logical errors
18.	Write a Python class to reverse a string word by word	Write a program to reverse the given string	38. Build the application 39. Develop the logic for string concatenation and substring 40. Rectify the syntax errors 41. Debug logical errors
19.	Write a program to: Create a list, add element to list, delete element from the lists.	Write a Python program to identify various lists and list manipulation methods in Python.	42. Create a one list with correct syntax 43. Create a list 44. Read elements from list 45. Add elements to list 46. Delete elements 47. Rectify the syntax errors 48. Debug logical errors 49. Check for the correctness of output for the given input
20.	Write a program to: Sort the list, reverse the list and counting elements in a list.	Write a Python program to identify various lists and list manipulation methods in Python.	50. Create a one list with correct syntax 51. Create a list 52. Read elements from list 53. Add elements to list 54. Delete elements 55. Rectify the syntax errors 56. Debug logical errors 57. Check for the correctness of output for the given input
21	Write a program to demonstrate working with tuples in python	Write a Python program to identify various tuples and manipulate	58. Create a tuple with correct syntax 59. Create a tuple 60. Read elements from tuple 61. Add elements to tuple 62. Delete elements 63. Rectify the syntax errors 64. Debug logical errors 65. Check for the correctness of output for the given input
	Write a program to: Create dictionary, add element to dictionary, delete element from the dictionary.	Write a Python program to identify various dictionary and dictionary	66. Create a one dictionary with correct syntax 67. Create a dictionary

22.		manipulation methods in Python.	68. Read elements from list 69. Add elements to dictionary 70. Delete elements from dictionary 71. Rectify the syntax errors 72. Debug logical errors 73. Check for the correctness of output for the given input
23	Write a program to: To calculate average, mean, median, and standard deviation of numbers in a list.	Write a Python program to identify various statistical functions.	74. Create a list 75. add elements to list 76. perform statistical functions on that list
24.	File Input/output: Write a program to: i) To create simple file and write "Hello World" in it. ii) To open a file in write mode and append Hello world at the end of a file.	Write a Python program to identify the steps to create a file and append to file.	77. Create a Python file 78. Add contents to file
25	Write a program to: i) To open a file in read mode and write its contents to another file but replace every occurrence of character 'h' ii) To open a file in read mode and print the number of occurrences of a character 'a'.	Write a Python program to identify the steps to open a file in read/write mode.	79. Open a Python file in write mode 80. Add contents to the file 81. Open a Python file in Read mode 82. Print the file
26.	Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order	Write a Python program to identify the steps to open a file in read/write mode.	83. Open a Python file in write mode 84. Add contents to the file 85. Open a Python file in Read mode 86. Develop the logic to find unique words 87. Print the unique words
27.	Write a Program to: Add two complex number using classes and objects.	Write a Python program to identify the steps to create class and create an object in Python.	88. Create a class using Python 89. Create an object in Python 90. Debug the python program 91. Check the correctness
28	Write a Program to: Subtract two complexes number using classes and objects	Write a Python program to identify the steps to create class and create an object in Python.	92. Create a class using Python 93. Create an object in Python 94. Debug the python program Check the correctness
29	Write a pandas program to load the CSV into a dataframe and perform arithmetic operations on the data loaded	Write a Python program using the pandas library to perform the task	95. Import the pandas python library 96. Write the python code to load the CSV into dataframe 97. Perform the arithmetic operations 98. Debug the python program 99. Check the correctness
30.	Write a program to create a 5 * 5 Numpy	Write a Python program	100. Import the Numpy

	array with random integers between 1 and 100	using the Numpy library to perform the task	python library 101. Write the python code to create an array of size 5 *5 102. Generate the random integers between 1 and 100 103. Debug the python program 104. Check the correctness

Table specifying the scope of syllabus to be covered for unit tests

Unit Test	Learning outcomes to be covered
Unit test-1	From 1 to 10
Unit test-2	From 11 to 20
Unit test-3	From 21 to 30

COMPUTER HARDWARE LAB

Course Code	Course title	No of periods/ week	Total no of periods	Marks for FA	Marks for SA
26CM108L	Computer Hardware Lab	4	120	40	60

TIME SCHEDULE:

S No	Chapter/ Unit Title	No. of Periods	COs Mapped
1.	Computer Hardware Basics	24	CO1
2.	Mother Board Basics	24	CO2
3.	Hard Disk Basics	24	CO3
4	Operating System Basics	28	CO4
5	Troubleshooting Basics	20	CO5
	Total	120	

Course Objectives:

COURSE OBJECTIVES	1. Identify all the components of Computer. 2. Identify all the components of mother board. 3. Identify Different Types of Connectors & Pins on mother board. 4. Know Various options in CMOS setup 5. Assemble & Dis-Assemble a computer 6. Install RAM, CPU, HDD 7. Partition & format hard drive 8. Install system software's & printers 9. Identify all the settings of control panel 10. Identify the voltages in SMPS with color coding 11. Troubleshoot hardware issues

Course Outcomes

Course Outcomes	CO1	CM108L.1	Identify and Assemble the PC with suitable components.
	CO2	CM108L.2	Identify the components, connectors & pins on a Motherboard
	CO3	CM108L.3	Install and partition the Hard disk
	CO4	CM108L.4	Install operating system and printers
	CO5	CM108L.5	Troubleshoot hardware devices

Learning Outcomes:

Computer Hardware Basics

1. Identification of various Hardware components of Computer
2. Using various options of CMOS setup
3. Installing and removing of Motherboard from computer
4. Printing the summary of your system Hardware and verify for correctness
5. Assembling of a PC
6. De-Assembling of a PC

Mother Board Basics

7. Identification of various components of Motherboard
8. Identification of various back panel connectors of Motherboard
9. Identification of various pins & connectors
10. Installing and removing RAM, CPU, HARD DISK
11. Identification of various motherboards based on the form factor such as AT, ATX, micro ATX, mini ATX , Baby AT et
12. Identification on voltage levels of each wire in SMPS based on the standard color of the wire.

Hard Disk basics

13. Familiarity with hard disk interfacing standards like IDE/SCSI /SATA / PATA
14. Practice on Partition of Hard disk
15. Practice on formatting a hard disk in FAT/NTFS Format
16. Calculating Hard Disk Storage Capacity
17. Recovery of lost data on hard drive using Recovery Tools
18. Defragmenting the Hard Disk

Operating system basics

19. Installation of operating system software (Windows 10/11)
20. Installation of device driver software
21. Installation of application software (MS-Office 2007/ 2010)
22. Practice on various options of control panel
23. Installation of printer using control panel
24. Installation of any anti-virus software for protecting pc
25. Practice on changing resolution, color, appearance, screensaver options of the display Setting in pc
26. Knowing the precautions to be taken while troubleshooting the hardware

Trouble shooting Basics

27. Identification of steps in troubleshooting: Visual inspection, Layman checks, measurement of voltage levels, Beep sounds, Error codes and Use of Advanced Diagnostic tools
28. Troubleshooting Keyboard, Printer Problems
29. Troubleshooting no display on monitor Problem
30. Troubleshooting RAM Issues

CO-PO/PSO MATRIX

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM108L.1	1	0	1	2	1	0	2	3	0	0
CM108L.2	1	0	2	2	1	0	1	3	1	0
CM108L.3	1	0	2	1	0	0	1	2	2	1
CM108L.4	1	0	2	1	0	0	1	2	2	1
CM108L.5	1	0	2	1	0	0	1	2	2	1
Average	1	0	1.8	1.4	0.4	0	1.2	2.4	1.4	0.6

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

S. No.	Name of the experiment	Objectives	Key Competencies
1	Exercise on Identification and familiarization of various components of computer system.	Identification and familiarization of various components of computer system.	❖ Identify and note down mother board, Components and Chips. ❖ Identify various Internal and External slots in the mother board and clean them with blower/ Brush. ❖ Practicing Inserting and Removing RAM with care.
2	Exercise on various operations and modifications required for CMOS setup.	Perform various operationsand modifications required for CMOS setup.	❖ Identify location of CMOS battery on mother board. ❖ Know how to replace CMOS battery. ❖ Identify keyboard key for entering BIOS setup. ❖ Setup CMOS settings ❖ Check the status of CMOS settings after replacement.
3	Exercise on Installing and removing of Motherboard from computer	Perform mother board installation, removal and identification	❖ Identify motherboard form factors (ATX, microATX, mini-ITX) ❖ Recognize various motherboard components (CPU socket, RAM slots, chipset, PCIe slots, power connectors) ❖ Demonstrate correct procedure for removing and installing a motherboard ❖ Properly mount the motherboard using standoffs and screws ❖ Align the I/O shield and rear ports correctly

4	Exercise on Print the summary of your system Hardware and verify for correctness	Print the summary of your system Hardware and verify for correctness	❖ Know how to open system summary window ❖ Check whether all the hardware peripherals are working properly or not. ❖ Know how to install device drivers ❖ Know how to enable and disable hardware peripherals. ❖ Print the hardware summary page
5	Assembling a PC	understand the step-by-step process of assembling a personal computer.	❖ Recognize essential hardware components (motherboard, RAM, CPU, HDD, PSU, cabinet, etc.) ❖ Understand the role and placement of each component ❖ Install the motherboard, processor, RAM, and storage devices into the cabinet ❖ Mount and connect the SMPS ❖ Connect cables (SATA, power, front panel I/O, etc.) ❖ Install expansion cards (optional) ❖ Connect and manage power cables safely and efficiently ❖ Organize internal wiring to promote airflow and safety
6	De-Assembling of a PC	To understand the systematic process of disassembling a personal computer.	❖ Identify all major components: motherboard, CPU, RAM, HDD/SSD, SMPS, GPU, etc. ❖ Detach cables and connectors carefully (SATA, power, front panel, etc.) ❖ Use tools like screwdrivers correctly and safely. ❖ Handle delicate components such as RAM and processor without damage. ❖ Follow a correct and logical sequence for disassembling the system. ❖ Place screws and components in organized manner for easy reassembly.
7	Identification of various components of Motherboard	To identify and understand the function of key components on a computer motherboard.	❖ Identify components such as CPU socket, RAM slots, chipset, power connectors, SATA/IDE ports, CMOS battery, etc. ❖ Recognize expansion slots (PCI, PCIe, AGP), and onboard headers (USB, front panel, fan headers). ❖ identify the purpose of each component (e.g., chipset controls data flow, CPU socket holds the processor). ❖ Understand how components work together within the computer system.
8	Identification of various back panel connectors of Motherboard	To identify the different connectors present on the back panel of a motherboard.	❖ Recognize common back panel ports: • **USB Ports** (Type-A, Type-C) • **HDMI / VGA / DVI** • **Ethernet (RJ-45)** • **Audio Jacks** • **PS/2 Ports** • **DisplayPort / Serial / Parallel Ports** (if present) ❖ identify the use of each connector for devices like monitors, keyboards, mice,

			network cables, printers, speakers, etc. ❖ Visually and physically distinguish between similar-looking ports (e.g., USB vs. HDMI).
9	Identification of various pins of connectors in Motherboard	To identify different types of connector pins and headers located on the motherboard.	❖ Recognize key pin headers such as: ❖ Back panel connectors (PS2, USB, HDMI, VGA,DVI,RJ-45,SERIAL,PARALLEL PORTS) ❖ **Front panel connector (F_PANEL):** power switch, reset switch, power LED, HDD LED ❖ **USB headers (USB1, USB2...)** ❖ **Audio headers (HD_AUDIO, AC'97)** ❖ **Fan headers (CPU_FAN, SYS_FAN)** ❖ **Power connectors (24-pin ATX, 4/8-pin CPU power)** ❖ **SATA ports and M.2 slots** ❖ Understand the layout of pins (e.g., +ve, -ve, GND, VCC) for each connector. ❖ Read diagrams and pin out charts to match wires correctly.
10	Installing and removing RAM,CPU,HARD DISK in motherboard	To learn the correct procedure for installing and removing **RAM, CPU, and Hard Disk** in a computer system.	❖ Properly insert and remove RAM modules from DIMM slots. ❖ Correctly install the CPU into the socket (LGA/PGA) and apply thermal paste if required. ❖ Install and connect SATA/SSD hard disks, including power and data cab

11	Identify various motherboards based on the form factor such as AT,ATX, micro ATX, mini ATX , Baby AT etc	To identify and differentiate various motherboard form factors used in personal computers.	❖ Visually identify the key characteristics of common form factors such as: • **AT (Advanced Technology)** • **ATX (Advanced Technology Extended)** • **Micro ATX** • **Mini ATX / Mini-ITX** • **Baby AT** ❖ Understand how form factor affects: • Cabinet size and type • Power supply connector types • Number of expansion and memory slots • Cooling options
12	Identify voltage levels of each wire in SMPS based on the standard color of the wire	To identify different wires of the SMPS (Switched Mode Power Supply) based on their color codes.	❖ Know the purpose and acceptable range of each voltage rail. ❖ Understand potential issues from overvoltage or under voltage. ❖ Learn how to safely measure voltage from an SMPS using a multimeter. ❖ Follow ESD and electrical safety protocols while dealing with power supply wires. ❖ Match wire voltages to the components they power (e.g., CPU requires 12V, USB uses 5V). ❖ Identify different SMPS connectors (24-pin ATX, 4-pin CPU, SATA, Molex) and their pin/wire color mapping.
13	Familiarize with hard disk interfacings standards like IDE/SCSI /SATA / PATA	To understand various types of hard disk interfaces used in personal computers	❖ Identify and differentiate between IDE, PATA, SATA, and SCSI interfaces by their connectors and cables. ❖ Understand the technical specifications (speed, number of pins, data transfer type) of each interface standard. ❖ Recognize compatible hard disks and motherboards for each interface type. ❖ Understand the historical development and obsolescence of older standards (IDE, PATA) and the adoption of newer ones (SATA).
14	Practice on Partition of Hard disk	To understand the concept and purpose of hard disk partitioning	❖ Grasp how a hard disk can be divided into multiple partitions for better data management. ❖ Use system tools like Windows Disk Management or diskpart (command line) for partitioning tasks. ❖ Choose appropriate file systems (e.g., NTFS for Windows) during formatting. ❖ Apply logical planning to create partitions based on user/system needs (e.g., OS, data, backup)

15	Practice on formatting a hard disk FAT/NTFS Format	To understand the concept and purpose of formatting a hard disk	❖ Distinguish between FAT32 and NTFS in terms of features, limitations, and usage scenarios. ❖ Format a hard drive or partition using both graphical and command-line tools (e.g., format, diskpart). ❖ Select the appropriate file system based on the system requirement (e.g., NTFS for Windows, FAT32 for external drives). ❖ Configure volume labels and choose cluster sizes based on the type of usage.
16	Calculating Hard Disk Storage Capacity	To understand the components that determine a hard disk's storage capacity	❖ Understand disk structure: **number of platters**, **tracks**, **sectors**, **bytes per sector**, etc. ❖ Convert between different units of storage (e.g., 1 GB = 1024 MB, etc.) ❖ Interpret technical specs of a hard drive (e.g., from a product label or manual) to estimate usable capacity. ❖ Understand why **usable space is less than labeled capacity** (e.g., due to formatting, file system overhead, 1 GB = 1000 MB marketing vs. 1024 MB actual).
17	How to recover lost data on hard drive using Recovery Tools	How to recover lost data on hard drive.	❖ Verify the available recovery tools ofOperating system. ❖ Know how to recover lost data on Harddrive using Restore point. ❖ Know how to recover lost data on Harddrive using Recovery Image.
18	Defragmenting the Hard Disk	To understand the concept of disk fragmentation and how it affects system performance.	❖ Identify how data gets fragmented on traditional hard drives and its impact on read/write speed. ❖ Use built-in tools like **Windows Defragment and Optimize Drives**. ❖ Analyze disk drives for fragmentation percentage before initiating defragmentation. ❖ Perform defragmentation safely without interrupting other operations. ❖ Observe changes in performance post-defragmentation. ❖ Set up scheduled defragmentation for regular maintenance.
19	Installation of operating system software (Windows 10 / 11)	To understand the step-by-step procedure of installing a Windows operating system.	❖ Prepare a PC for OS installation (bootable media, BIOS settings, partitioning) ❖ Carry out a complete and clean installation of Windows 10 or Windows 11. ❖ Set and modify the boot device priority in BIOS/UEFI settings. ❖ Create, delete, and format partitions during installation as needed. ❖ Choose correct file system (e.g., NTFS) for Windows installation ❖ Install required drivers post-OS installation for full hardware support.

20	Installation of device driver software	To understand the role and importance of device drivers in a computer system.	❖ Recognize which devices require drivers (e.g., graphics card, audio, LAN, chipset). ❖ Use Windows Device Manager to view device status, update, install, or uninstall drivers. ❖ Install drivers from CDs, USBs, or downloaded packages. ❖ Fix common issues such as "driver not recognized," "yellow warning sign," or "code 10 error." ❖ Match correct driver versions to operating systems and hardware models. ❖ Test devices after driver installation to ensure proper working (e.g., sound output, network connectivity).
21	Installation of application software (MS-Office 2007/ 2010)	To install Microsoft Office 2007 / 2010 suite including Word, Excel, PowerPoint, and other components.	❖ Perform installation from CD/DVD, USB, or downloaded setup file. ❖ Choose between typical, custom, or complete installations depending on user needs. ❖ Check system compatibility (OS, RAM, storage) before installation ❖ Enter product key and complete activation process for licensed software. ❖ Choose and manage installation paths or directories during setup. ❖ Confirm installation by launching Office applications and checking version details.
22	Practice on various options of control panel	To explore and understand the purpose of various tools and settings available in the Windows Control Panel.	❖ Navigate through Control Panel efficiently in both Category and Classic views. ❖ Adjust screen resolution, theme, desktop background, and screen saver. ❖ Create, delete, and manage user accounts and passwords. ❖ Add and remove hardware devices; access printer settings. ❖ View and change adapter settings, connect to Wi-Fi, and set up local networks. ❖ Set system date/time, region, and language preferences. ❖ Install/uninstall programs, manage default apps, and check installed updates.
23	Installation of printer using control panel	To understand how to install a printer manually using the Control Panel in Windows	❖ Add a new printer through Control Panel using "Devices and Printers" > "Add a printer" option. ❖ Select or install appropriate drivers for USB, network, or wireless printers. ❖ Choose correct ports (USB, LPT1, TCP/IP) during manual installation. ❖ Set the installed printer as the default printer and configure preferences ❖ Identify and fix installation errors like “printer not found” or “driver unavailable.” ❖ Perform a successful test print to verify installation.

24	Installation of any anti-virus software for protecting pc	To understand the need for anti-virus software in protecting a computer from malware and other threats.	❖ Install anti-virus software from a setup file or online installer (e.g., Avast, Quick Heal, AVG, etc.). ❖ Grant necessary permissions during installation and accept license agreements. ❖ Run full system scans, quick scans, or custom scans and interpret the results. ❖ Identify threats, view quarantine folder, and restore or delete infected files. ❖ Ensure virus definitions are up to date and software is working effectively. ❖ Confirm background protection is active and responding to potential threats.
25	Practice on changing resolution, color, appearance, screensaver options of the display setting of pc	To understand how to access and modify display settings in a Windows PC.	❖ Access the Display Settings through the desktop or Control Panel. ❖ Change the resolution to optimize screen display based on monitor capability. ❖ Modify screen layout (landscape/portrait) as required. ❖ Choose appropriate color schemes and Windows themes for better visibility and aesthetics. ❖ Select, preview, and set timers for screensavers; configure password protection on resume. ❖ Use high contrast modes or larger text options for improved accessibility. ❖ Personalize display options according to user needs or preferences.
26	Know the precautions to be taken while troubleshooting the hardware	To understand the importance of safety and preventive measures during hardware troubleshooting.	❖ Recognize the dangers of working with powered devices and exposed circuitry. ❖ Use of anti-static wrist straps, mats, and ESD-safe tools to prevent component damage. ❖ Ensure power is disconnected before opening the cabinet or removing parts. ❖ Select and use appropriate tools like screwdrivers, tweezers, and testers correctly. ❖ Maintain a clean, dry, and well-lit workspace free from conductive materials. ❖ Avoid touching circuit traces or connectors with bare hands; handle by edges.
27	identify the systematic steps in troubleshooting: Visual inspection, Layman checks, measurement of voltage levels, Beep sounds, Error codes and Use of Advanced Diagnostic tools	To understand the step-by-step process of hardware troubleshooting.	❖ Identify burnt components, loose wires, broken connectors, or dust accumulation. ❖ Perform basic checks like verifying power supply, monitor connection, and cable tightness. ❖ Use a multimeter to measure voltage levels from the SMPS to motherboard and components (e.g., +12V, +5V, +3.3V lines). ❖ Interpret BIOS beep codes to identify faults in RAM, CPU, GPU, or motherboard ❖ Read and understand on-screen or LED error codes shown during POST. ❖ Use hardware diagnostic software (like Hiren's Boot CD, MemTest86, etc.) and built-in Windows tools for checking hardware health.
28	Trouble shooting Keyboard, Printer problems	To identify and diagnose common hardware and connectivity issues in keyboards and printers.	❖ Recognize issues like non-responsive keys, stuck keys, or full keyboard failure. ❖ Verify USB/PS2 port connection, wireless keyboard pairing, and power supply (for wireless).

			❖ Access and update keyboard drivers using Device Manager or Control Panel. ❖ Replace with a known working keyboard to verify device-level failure. ❖ Verify cables, network/Wi-Fi connectivity, and power status of the printer. ❖ Clear print queue, restart the print spooler, and reinstall printer software. ❖ Install or update appropriate printer drivers. ❖ Detect and resolve issues like paper jams, low ink/toner, or incorrect paper tray selection. ❖ Check printer settings in Control Panel or Settings and configure defaults properly. ❖ Use printer self-test or built-in diagnostic tools for hardware faults.
29	Trouble shooting no display on monitor	To understand the possible causes behind a blank or no-display monitor issue.	❖ Detect loose cables, damaged connectors, or physically broken monitor parts. ❖ Check VGA/HDMI/DisplayPort/DVI cables, power supply, and monitor input source selection. ❖ Confirm monitor power using power LED indicators or multimeter readings. ❖ Interpret beep codes indicating RAM or graphics card issues. ❖ Remove and re-seat RAM and GPU; test with alternate RAM/monitor where possible. ❖ Connect the system to a different monitor to isolate the issue (monitor vs. system).
30	Troubleshooting RAM Issues	To understand the symptoms and causes of RAM-related problems in a computer system.	❖ Identify RAM issues through signs like continuous beeping, blue screen errors, system freezing, or failure to boot. ❖ Detect physical damage, dust, or improper installation of RAM modules. ❖ Use BIOS/UEFI beep codes to identify RAM-related errors during POST (Power-On Self-Test). ❖ Safely remove, clean, and re-insert RAM sticks into the motherboard slots. ❖ Use tools like **MemTest86**, **Windows Memory Diagnostic**, or other bootable utilities to check RAM health. ❖ Test each RAM stick in different slots to identify faulty RAM or slot.

Table specifying the scope of syllabus to be covered for unit tests

Unit Test	Learning outcomes to be covered
Unit test-1	From 1 to 10
Unit test-2	From 11 to 20
Unit test-3	From 20 to 30

PHYSICS & CHEMISTRY LAB

Course Code	Course title	No. of Periods/week	Total no of periods	Marks For FA	Marks For SA
26CM109L	Physics & Chemistry Lab	3	45	20	30

PHYSICS LAB

Course Code	Course title	No. of Periods/week	Total no of periods	Marks For FA	Marks For SA
26CM109L	Physics Lab	3	45	20	30

TIME SCHEDULE

S. No	List of experiments	No. of Periods	COs
1.	Vernier calipers	03	CO1
2.	Micrometer (Screw gauge)	03	
3.	Verification of Lami's theorem using concurrent forces	03	
	Revision	03	
4.	Determination of 'g' using simple pendulum	03	CO2
5.	Focal length and focal power of convex lens by distant object method and U-V method	03	
6.	Verification of Boyle's law using Quill tube	03	
	Revision	03	
7.	Drawing of magnetic lines of force	03	CO3
8	Resonance apparatus–Determination of velocity of sound in air	03	
9	Refractive index of a solid using travelling microscope	03	
	Revision	03	
	Experiments for demonstration		
10	Meter bridge–Determination of resistance and specific resistance of material of given wire	03	CO4
11	Projectile motion- study the range of a projectile for different launch angles	03	
12	Generation of Beats using water columns	03	
	Total:	**45**	

COURSE OBJECTIVES

Upon completion of the course the student shall be able to

(1)	Apply practical physics principles to operate, troubleshoot, and optimize engineering devices.
(2)	Develop scientific skills through designing, conducting, and evaluating industry-relevant experiments to enhance technical proficiency.

COURSE OUTCOMES

CO1	CI109.1	Apply measurement techniques to improve accuracy; Explain forces maintaining equilibrium in physical systems.
CO2	CI109.2	Determine acceleration due to gravity experimentally; Investigate refraction of light at curved surfaces; Relate the gas pressure to volume variations at constant temperature.
CO3	CI109.3	Analyze the combined effect of magnetic fields (Earth and artificial magnet); Determine velocity of sound in air using resonance; Demonstrate U-V method to understand the refraction of light at curved surfaces.
CO4	CI109.4	Apply Kirchhoff's laws to compute the resistivity of a wire; Examine the projectile motion parameters; Observe and Interpret beat generation phenomenon.

LEARNING OUTCOMES

Upon completion of the course the student shall be able to

1. **Apply** measurement techniques using Vernier Calipers to **determine** the volumes of a cylinder and a sphere.
2. **Use** a screw gauge to **measure** and **determine**the thickness of a glass plate and the cross-sectional area of a wire.
3. **Verify** Lami's Theorem by **analyzing** a system of concurrent forces.
4. **Conduct** simple pendulum experiment to **calculate** the acceleration due to gravity (g) and **interpret** the result through an L–T^2 graph.
5. **Determine** the focal length and power of a convex lens using distant object method and U–V method, and **compare** the results.
6. **Verify** Boyle's Law using a Quill tube by noting pressure (P) and length of air column(L).
7. **Illustrate** the behaviour of lines of magnetic field around a bar magnet using magnetic compass.

8. **Determine** the velocity of sound in air at room temperature and at 0°C using resonance apparatus.
9. **Determine** the refractive index of a solid by **using**the measurements taken with a travelling microscope.
10. **Demonstrate** the use of a meter bridge to **determine** the resistance and specific resistance of a given wire.
11. **Simulate** projectile motion and **observe** the range of the projectile for different launch angles using appropriate experimental setup.
12. **Demonstrate** the phenomenon of beats by **creating** beat patterns using water columns.

CO-PO MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	1	1	1	1	1	1
CO2	3	1	1	1	1	1	1
CO3	3	1	1	1	1		1
CO4	3	1	1	2	1		1

3 = strongly mapped, 2 = moderately mapped, 1 = slightly mapped

CO-PO MAPPNG STRENGTH

Common-CI109	**Physics Lab**				**No of periods: 90**
POs	Mapped with CO No	CO periods addressing PO in Column 1		Level 1,2,3	Remarks
		No	%		
PO1	CO1,CO2,CO3,CO4	15	33.3%	2	>40% level 3 (highly addressed)
PO2	CO1,CO2,CO3,CO4	6	13.3%	1	
PO3	CO1,CO2,CO3,CO4	5	11.1%	1	25% to 40% level2 (moderately addressed)
PO4	CO1,CO2,CO3,CO4	6	13.3%	1	
PO5	CO1,CO2,CO3,CO4	5	11.1%	1	5% to 25% level1 (Low addressed)
PO6	CO1,CO2	3	6.7%	1	
PO7	CO1,CO2,CO3,CO4	5	11.1%	1	<5% (not addressed)

3 = strongly mapped, 2 = moderately mapped,1 = slightly mapped

Note: The gaps in CO and PO mapping will be achieved by one or more appropriate activities from the following.

(i) Seminars (ii) Viva-voce (iii) Assignments (iv) Quiz competitions (v) Industrial visits (vi) Tech fest (vii) Mini project (viii) Group discussions (ix) Virtual labs (x) Library visit for e-books

COURSE CONTENT

Name of the Experiment	**Competencies (Revised Bloom's Taxonomy)**	**Key Competencies (Revised Bloom's Taxonomy)**
1. Practice on Vernier Calipers	• Determine the least count • Place the object in accurate position. • Interpret scale readings • Calculate volume of cylinder and sphere	• Interpret Vernier readings • Compute volume using appropriate formulae • Apply measurement data to calculate physical quantities
2. Practice on Screw Gauge	• Determine the least count and zero error • Place the object in accurate position. • Interpret scale readings • Calculate thicknessand cross-sectional area	• Analyze scale readings for zero error • Compute thickness and area from measurements • Apply micrometer data to solve practical problems
3. Verification of Lami's Theorem	• Setup experimental arrangement • Apply appropriate weights • Measure angles between forces • Analyze data to verify theorem	• Interpret directions and angles of forces • Evaluate force relationships • Validate Lami's Theorem using experimental data
4. Simple Pendulum	• Arrange the pendulum properly • Measure the time taken for 20 oscillations • Compute time period and acceleration due to gravity • Plot L–T^2 graph	• Measure oscillation intervals accurately • Calculate g using experimental data • Interpret L–T^2 graph to confirm relationship
5. Focal Length and Power of Convex Lens	• Place the object and convex lens in proper positions. • Measure image distance • Compute focal length and power	• Determine focal length using both methods • Validate optical formulae using experiment
6. Boyle's Law Verification	• Record atmospheric pressure • Measure air column length and calculate the enclosed pressure • Analyze data for PxL consistency	• Setup quill tube in different positions for multiple readings • Interpret pressure–length data
7. Drawing of Magnetic Lines of force	• Draw meridian and set magnet orientation • Sketch the lines of magnetic field using compass.	• Visualize field pattern accurately • Analyze field symmetry

8. Velocity of Sound – Resonance Method	• Assemble apparatus and adjust reservoir • Identify resonating lengths • Calculate velocity of sound at room temperature and at $0^{o}C$.	• Detect resonance points • Compute velocity using resonance data • Extrapolate to standard temperature
9. Refractive Index of a solid using Traveling Microscope	• Determine least count • Measure real and apparent thickness • Calculate refractive index	• Analyze scale readings • Apply refraction formula • Interpret refractive index of a solid.
10. Meter Bridge	• Connect circuit properly • Measure balancing length, radius of given wire • Compute resistance and specific resistance	• Analyze circuit behavior • Calculate unknown resistance • Interpret experimental values for resistivity
11. Projectile motion- study the range of a projectile for different launch angles	• Setup and align launcher • Adjust launch angles • Measure range	• Observe the variations in horizontal range for different angles of projection. • Evaluate trajectory data
12. Generation of Beats using water columns	• Setup beat source using glasses or online tone generator • Generate close frequencies • Detect and analyze beat pattern	• Observe frequency interference • Interpret beat frequency data • Analyze patterns using mobile sensors/ software

SCHEME OF VALUATION FOR END PRACTICAL EXAMINATION

Activity	**Marks**
Aim, Apparatus, Formulae	6
Tabulations and Readings	12
Calculations	4
Precautions, Results	3
Viva-voce	5
Total marks	**30**

CHEMISTRY LAB

Course code	Course title	No. Of periods/week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM109L	CHEMISTRY LAB	3	45	20	30	2

TIME SCHEDULE

S. No.	Name of the Experiment	No. of Periods	COs Mapped
1.	Introduction to Fundamentals of Analytical Chemistry.	03	CO1
2.	Chemical Recognition by Sensory Cues.	03	CO1
3.	Preparation of Standard Na_2CO_3 Solution.	03	CO1
4.	Estimation of HCl Using Standard NaOH Solution.	03	CO2
5.	Determination of Alkalinity of Water Sample.	03	CO2
6.	Estimation of Mohr's Salt Using Standard $KMnO_4$ Solution.	03	CO3
7.	Determination of Total Hardness of Water Using Standard EDTA Solution.	03	CO4
8.	Estimation of Chlorides Present in Water Sample Using Standard $AgNO_3$ Solution.	03	CO4
9.	Analyzing pH of Common Compounds Using Visual and Instrumental Methods.	03	CO5
10.	Estimation of Copper Deposited on an Object by Using Electrolysis Process.	03	CO5
	Demonstration Experiments		
11.	Determination of Turbidity of Water Sample by Using Nephelometer.	03	CO5
12.	Construction and Working of Galvanic Cell.	03	CO5
13.	Preparation of a Polymer (Bakelite).	03	CO5
14.	Open Ended Experiments/Micro Projects – I.	03	CO5
15.	Open Ended Experiments/Micro Projects – II.	03	CO5
	TOTAL	**45**	

COURSE OBJECTIVES

Upon completion of the course the shall be able to:	
(i)	To Perform fundamental analytical chemistry techniques, identify chemical substances using sensory cues and accurately prepare standard solutions.
(ii)	To Evaluate and judge the neutralization point in acid base titration.
(iii)	To Evaluate the endpoint of reduction and oxidation reaction.
(iv)	To Judge the stable end point of complex formation, stable precipitation.
(v)	To Determine the pH of compounds, estimate copper using electrolysis, measure water turbidity, demonstrate the working of a galvanic cell and prepare a polymer.

COURSE OUTCOMES

CO1	CI109.1	Perform fundamental analytical chemistry techniques, identify chemical substances using sensory cues and accurately prepare standard solutions.
CO2	CI109.2	Evaluate and judge the neutralization point in acid base titration.
CO3	CI109.3	Evaluate the endpoint of reduction and oxidation reaction.
CO4	CI109.4	Judge the stable end point of complex formation, stable precipitation.
CO5	CI109.5	Determine the pH of compounds, estimate copper using electrolysis, measure water turbidity, demonstrate the working of a galvanic cell and prepare a polymer.

LEARNING OUTCOMES:

Upon completion of the course the student shall be able to:

1.0 Practice volumetric measurements (using pipettes, measuring jars, volumetric flask, burettes) and gravimetric measurements (using different types of balances), making dilutions, etc.

2.0 Identify the chemical compounds and solutions by senses.

3.0 Practice making standard Na_2CO_3 solutions.

4.0 Conduct titrations adopting standard procedures and using standard NaOH solution for estimation of HCl.

5.0 Conduct titrations adopting standard procedures to determine the alkalinity of given samples of water (one ground water and one surface / tap water) using standard H_2SO_4 solution.

6.0 Conduct titrations adopting standard procedures and using standard $KMnO_4$ solution for estimation of Mohr's Salt.

7.0 Conduct titrations adopting standard procedures to determine the total hardness of given samples of water (one ground water and one surface / tap water) using standard EDTA solution.

8.0 Conduct titrations adopting standard procedures to determine the chlorides present in the given samples of water (one ground water and one surface / tap water) and waste water by using standard $AgNO_3$ solution.

9.0 Conduct the test on given samples of water / solutions (like soft drinks, sewage etc.,) to determine their pH using pH paper, Universal indicator, digital pH meter.

10.0 Carry out the electrolysis process to estimate the amount of Copper deposited on an object.

11.0 Conduct the test on given samples of solutions (coloured and non-coloured) to determine their turbidity in NTU.

12.0 Understand the construction and working principle of a Galvanic cell and identify how chemical energy is converted into electrical energy through redox reactions.

13.0 Understand the process of polymerization by preparing Bakelite, a thermosetting polymer and study its formation through a condensation reaction.

14.0 Collect water sample from nearby water body and test for any two parameters.

15.0 Collect water sample from nearby sewage/industrial effluent and test for any two parameters.

[Note: Parameters – Alkalinity, Hardness, Chloride and pH]

CO – PO/PSO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PSO1**	**PSO2**	**PSO3**
CO1	3	2		1			1			
CO2	3	2		1						
CO3	3	2		1						
CO4	3	2		1	1					
CO5	3	2	1	1	1		1			
Average	**3**	**2**	**1**	**1**	**1**		**1**			

3 = Strongly Mapped 2 = Moderately Mapped 1 = Slightly Mapped

Note:

The gaps in CO and PO mapping will be achieved by one or more appropriate activities from the following:

(i). Assignments, (ii). Tutorials, (iii). Guest Lectures, (iv). Seminars, (v). Quiz Competitions, (vi). Industrial Visit, (vii). Tech Fest, (viii). Mini Projects, (ix). Group Discussions, (x). Virtual Classes and (xi). Library Visits.

COMPETENCIES AND KEY COMPETENCIES TO BE ACHIEVED BY THE STUDENT:

Name of the Experiment (No of Periods)	**Competencies**	**Key Competencies**
Introduction to Fundamentals of Analytical Chemistry. (03)	• Develop a foundational understanding of analytical chemistry principles and demonstrate proficiency in basic laboratory techniques, data analysis, and safety protocols.	• Students will master the foundational principles and laboratory techniques of analytical chemistry.
Chemical Recognition by Sensory Cues. (03)	• Develop skills in conducting simple tests and making accurate observations. • Interpret results to draw conclusions about the nature of chemical compounds.	• Develop skills in conducting simple tests and making accurate observations. • Interpret results to draw conclusions about the nature of chemical compounds.
Preparation of Standard Na_2CO_3 Solution. (03)	• Weighing the salt to the accuracy of 0.01mg • Measuring the water with volumetric flask, measuring jar, volumetric pipette and graduated pipette.	• Weighing the salt to the accuracy of 0.01 mg. • Measuring the water with volumetric flask, measuring jar, volumetric pipette and graduated pipette.
Estimation of HCl Solution Using Standard NaOH Solution. (03)	• Cleaning the glass ware and rinsing with appropriate solutions. • Making standard solutions • Measuring accurately the standard solutions and titrants. • Filling the burette with titrant. • Fixing the burette to the stand	• Making standard solutions. • Measuring accurately the standard solutions and titrants. • Effectively Controlling the flow of the titrant. • Identifying the endpoint. • Making accurate observations.
Determination of Alkalinity of Water Sample (03)		
Estimation of Mohr's Salt Using Standard $KMnO_4$ Solution. (03)		

Determination of Total Hardness of Water Using Standard EDTA Solution. (03) Estimation of Chlorides Present in Water Sample by Using Standard $AgNO_3$ Solution. (03)	• Effectively Controlling the flow of the titrant. • Identifying the endpoint. • Making accurate observations. • Calculating the results.	
Analyzing pH of Common Compounds Using Visual and Instrumental Methods. (03)	• Know pH range (0 – 14) and classify substances as acidic, neutral and basic. • Accurately measure pH using pH paper and universal indicator. • Note color changes and interpret pH values correctly. • Perform precise pH tests to ensure reliable results. • Record pH data and observations clearly. • Connect pH results to real-world contexts. • Familiarize with instrument. • Choose appropriate 'Mode' / 'Unit'. • Prepare standard solutions / buffers, etc. • Standardize the instrument with appropriate standard solutions. • Make measurements accurately.	• Accurately measure pH using pH paper and universal indicator. • Perform precise pH tests to ensure reliable results. • Prepare standard solutions/buffers, etc. • Standardize the instrument with appropriate standard solutions. • Make measurements accurately.
Estimation of Amount of Copper Deposited on an Object by Using Electrolysis Process. (03)	• Prepare standard solutions. • Selection of electrodes. • Set up and perform an electrolysis experiment accurately and safely. • Analyze experimental data to calculate deposition of	• Set up and perform an electrolysis experiment accurately and safely. • Analyze experimental data to calculate deposition of copper

Determination of Turbidity of Water Sample by Using Nephelometer. (03)	• Familiarize with instrument. • Choose appropriate 'Mode' / 'Unit'. • Prepare standard solutions / buffers, etc. • Standardize the instrument with appropriate standard solutions. • Make measurements accurately.	• Prepare standard solutions etc. • Standardize the instrument with appropriate standard solutions. • Make measurements accurately.
Construction and Working of Galvanic Cell. (03)	• Prepare standard solutions. • Selection of electrodes. • Making of salt bridge. • Construct a simple galvanic cell using appropriate electrodes and electrolyte solutions. • Explain the working principle of a galvanic cell, including electron flow, redox reactions, and the function of the salt bridge.	• Construct a simple galvanic cell using appropriate electrodes and electrolyte solutions. • Explain the working principle of a galvanic cell, including electron flow, redox reactions, and the function of the salt bridge.
Preparation of a Polymer (Bakelite). (03)	• Identify and handle the raw materials (phenol and formaldehyde) used in Bakelite preparation safely. • Perform the polymerization reaction under controlled conditions. • Observe and describe the physical changes during polymer formation. • Develop laboratory skills related to synthesis and handling of polymers.	• Perform the polymerization reaction under controlled conditions. • Develop laboratory skills related to synthesis and handling of polymers.

Open Ended Experiments/ Micro Projects – I. (03)	• Identifies a relevant chemical problem or question based on prior knowledge. • Demonstrates proficiency in basic chemistry lab techniques (e.g., titration, preparation). • Prepares solutions accurately (molarity, dilutions, standardizations). • Handles chemicals safely following MSDS guidelines and standard lab practices. • Uses instruments relevant to the experiment (e.g., pH meter, nephelometer and digital balance) correctly. • Calibrates and maintains instruments when needed. • Interprets instrumental output with understanding of underlying chemical principles. • Makes accurate, timely and detailed observations of chemical reactions (e.g., color changes, precipitate formation). • Records quantitative and qualitative data systematically. • Maintains an organized and complete lab notebook. • Works effectively in pairs or groups; shares responsibilities and discusses findings collaboratively. • Presents results through oral discussion, lab reports, or visual presentations using proper chemical terminology.	• Prepares solutions accurately (molarity, dilutions, standardizations). • Uses instruments relevant to the experiment (e.g., pH meter, nephelometer and digital balance) correctly. • Calibrates and maintains instruments when needed. • Makes accurate, timely and detailed observations of chemical reactions (e.g., color changes, precipitate formation).
Open Ended Experiments/ Micro Projects – II. (03)		

REFERENCE BOOKS

1. **VOGEL's Textbook of Quantitative Analysis, Sixth Edition, Pearson Education Limited.**
2. **VOGEL's Textbook of Qualitative Analysis, Seventh Edition, Pearson Education Limited.**
3. **Y. Bharathi Kumari & Jyotsna Cherukuri - Laboratory Manual of Engineering Chemistry for Engineering Students of JNT Universities.**
4. **Instrumental Methods of Chemical Analysis.**
5. **NCERT Chemistry Laboratory Manual for Class XII.**
6. **Practical Chemistry by the Royal Society of Chemistry Education.**

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TEST – I & UNIT TEST – II

Unit Test	Learning Outcomes to be Covered
Unit Test – 1	From Experiment 1 to 5
Unit Test – 2	From Experiment 6 to 10

COMPUTER & DIGITAL SKILLS LAB

Course code	Course Title	No. of Periods / Week	Total No. of periods	Marks for FA	Marks for SA
26CM110L COMMON TO ALL BRANCHES	COMPUTER & DIGITAL SKILLS LAB	3	90	40	60

Time schedule:

Chapter No.	Chapter / Unit Title	No. of sessions (3 periods per session)	No. of Periods	CO's Mapped
1.	Computer hardware and Software Basics	1	3	CO1
2.	Windows Operating System	1	3	CO1
3.	MS Word	6	18	CO2
4.	MS Excel	7	21	CO3
5.	MS PowerPoint	6	18	CO4
6	AI, ML & Quantum computing Tools	9	27	CO5
Total periods		**30**	**90**	

COURSE OBJECTIVES

COURSE OBJECTIVES	i) To familiarize with basics of Computer Hardware and Software ii) To familiarize with operating systems iii) To familiarize with Microsoft word iv) To familiarize with Microsoft Excel v) To familiarize with Microsoft PowerPoint vi) To familiarize with AI, ML, Quantum Computing Tools

Course outcomes

Course Outcomes	At the end of the course students will be able to		
	CO1	CM110L.1	Identify hardware and software components
	CO2	CM110L .2	Prepare documents with given specifications using word processing software
	CO3	CM110L .3	Use Spread sheet software to make calculations and to draw various graphs / charts.
	CO4	CM110L .4	Use Power point software to develop effective presentation for a given theme or topic.
	CO5	CM110L .5	To use basic AI, ML & Quantum Computing Tools

Learning Outcomes:

I. Computer Hardware and Software Basics

1. a). To get familiarized with Computer system and hardware connections
   b). To start and Shut down Computer correctly
   c). To explore Windows Desktop
2. To check the software details of the computer
3. To check the hardware present in your computer

II. Windows's operating system

4. To work with Files and Folders
5. To use Windows Accessories: Calculator – Notepad – WordPad – MS Paint

III. MS-WORD

6. To get familiarized with Ribbon layout of MS Word.
7. To perform basic word processing
8. To use basic formatting techniques
9. To insert a table of required number of rows and columns
10. To insert Objects, Clipart and Hyperlinks
11. To use Mail Merge feature of MS Word
12. To use Equations and symbols features

IV. MS-EXCEL

13. To get familiarized with MS-EXCEL ribbon layout
14. To access and enter data in the cells
15. To edit a spread sheet- Copy, Cut, Paste, and selecting Cells
16. To use built in functions and Data Formatting
17. To create Excel Functions, use autofill feature
18. To enter a Formula for automatic calculations
19. To sort and filter data in sheet.
20. To present data using Excel Graphs and Charts.

21. To format a Worksheet in Excel for printing using Page layout
22. To develop lab report formats of respective discipline.

V. Practice with MS-POWERPOINT

23. To get familiarized with Ribbon layout features of PowerPoint.
24. To create a simple PowerPoint Presentation
25. To set up a Master Slide in PowerPoint
26. To insert Text and Objects
27. To insert Flow Charts
28. To insert Tables
29. To insert Charts/Graphs
30. To insert video, audio & hyperlinks
31. To animate text, objects and slides.
32. To Review Presentations

VI. AI, ML & Quantum Computing Tools

33. To get familiarized with AI Tools
34. To get familiarized with working of ChatGPT
35. Identify Objects using AI Tools based on CNN, YOLO, SSD, R-CNN
36. To paraphrase text using AI Tools (PEGASUS, GPT, T5)
37. To use text-to-Image Generation AI Tools (DALL-E, MIDJOURNEY)
38. To use voice command simulation AI Tools (SPEECH-TO-TEXT)
39. To get familiarized with ML Tools
40. To get familiarized with Quantum Computing Tools
41. To familiarize with quantum bits (qubits) using Dirac notation
42. To familiarize the behaviour of single and multiple qubit gates.
43. To familiarize with Qubit as a Coin / Spin Analogy

CO-PO/PSO MATRIX

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM110L.1	**1**	**2**	**1**	**1**	**2**	**1**	**2**	**3**	**1**	**1**
CM110L.2	**2**	**2**	**3**	**3**	**2**	**1**	**2**	**3**	**1**	**1**
CM110L.3	**2**	**2**	**3**	**3**	**2**	**1**	**2**	**3**	**3**	**3**
CM110L.4	**2**	**2**	**3**	**3**	**2**	**1**	**2**	**3**	**2**	**1**
CM110L.5	**3**	**3**	**2**	**3**	**3**	**3**	**3**	**3**	**3**	**3**
Average	**2**	**2.2**	**2.4**	**2.6**	**2.2**	**1.4**	**2.2**	**3**	**2**	**1.8**

3=Strongly mapped, 2=moderately mapped, 1=slightly mapped

Key competencies:

Exp /Task/Ex No	Name of Experiment /Task /Exercise	Objectives	Key competencies
1 (a).	To get familiarized with Computer system and hardware connections	a. Connect cables to external hardware and operate the computer	a. Identify the parts of a computer system: i). CPU ii). Mother Board iii) Monitor iv)CD/DVD Drive v) Power Switch vi) Start Button vii) b. Reset Button viii) RAM ix) SSD /HDD c. Identify and connect various peripherals d. Identify and connect the cables used with computer system e. Identify various ports on CPU cabinet and connect Keyboard, Mouse and peripherals
1 (b).	Start and Shut down Computer correctly	a. Login and logout as per the standard procedure b. Operate mouse & Key Board	a. Log in using the password b. Start and shut down the computer c. Use Mouse and Key Board
1 (c).	Explore Windows Desktop	a. Access application programs using Start menu b. Use taskbar and Task manager	a. Familiarity with Start Menu, Taskbar, Icons and Shortcuts b. Access application programs using Start menu, Task manager c. Use Help support
2.	Check the software details of the computer System	a. Access the properties of computer and to find the details	a. Finding the details of operating system being used b. Finding the details of edition/version ServicePack installed
3.	Check the hardware present in your computer	a. Access device manager and to find the details b. Type /Navigate the correct path and Select icon related to the details required	a. Finding the CPU name andclock speed b. Finding the details of RAM andhard disk present c. Accessing Device manager using Control Panel and check the status of devices like mouse and key board d. Using My Computer to check the details of Hard drives andpartitions
4.	Working with Files and Folders	a Create files and folders b Rename, arrange and search for the required folder/file c. Restore deleted files from Recycle bin	a. Create folders and organize files in different folders b. Use cut, copy and paste commands to organize files and folders c. Arrange icons by name, size, type and Modified d. Search for a file or folder andfind its path e. Create shortcut to files and folders (in other folders) on Desktop

			f. Familiarity with the use of My Documents g. Familiarity with the use of Recycle Bin
5.	Use Windows Accessories like Calculator – Notepad –WordPad – MS Paint	a. Use windows accessories and select correct text editor based on the situation. b. Use MS paint to create /Edit picturesand save in the required format	a. Access Calculator using Run command b. Familiarity with the use ofCalculator c. Create Text Files using Notepad, WordPad and observe the difference in file sizes d. Use MS paint to create .jpeg, .bmp files
6.	Get familiarized with Ribbon layout of MS word.	a. Create a Document and name appropriately and save it b. Set paper size and print options	a. Create/Open a document b. Use Save and Save asfeatures c. Work on two Word documents simultaneously d. Choose correct Paper size and Printing options
7.	Perform basic Word Processing	a. Use key board and mouse to enter/edit text in the document. b. Use shortcuts c. use Spell / Grammar Check features for auto corrections	a. Typing text b. Keyboard usage c. Mouse Usage (Left click / Rightclick / Scroll) d. Using Keyboard shortcuts e. Using Find and Replace features in MS- word f. Use Undo and Redo Features g. Use spell check to correct Spellings and Grammar
8.	Use basic formatting techniques	a. Format Text and paragraphs and using various text styles. b. Use bullets and numbers to create lists. c. Use Templates /Themes d. Insert page numbers, date, headers and footers	a. Formatting Text b. Formatting Paragraphs c. Setting Tabs d. Formatting Pages e. Use various Font Styles f. Insert bullets and numbers g. Using Themes and Templates h. Insert page numbers, header and footer
9.	Insert a table of required number ofrows and columns	a. Insert table in the word document and edit b. Use sort option for arranging data.	a. Editing the table by adding the fields, deleting rows and columns, inserting sub table, marking borders. Merging and splitting of cells in a Table b. Changing the background color of the table c. Using table design tools d. Using auto fit – fixed row/ column height/length – Even distribution of rows / columns feature e. Converting Text to table and Table to Text f. Use Sort feature of the Table to arrange data in ascending/ descending order
10.	Insert objects, clipart and Hyperlinks	a. Insert hyperlinks & Bookmarks b. Create organization charts/flow charts	a. Creating a 2-page document and Insert hyperlinks and Bookmarks. b. Creating an organization chart c. Preparing an Examination schedule notice with a hyperlink to Exam schedule

			table.
11.	Use Mail merge feature of MS Word	Using Mail mergefeature	a. Using mail merge to prepare individually addressed letters b. Using mail merge to print envelopes.
12.	Use Equations and symbols features.	Enter Mathematical symbols and Equations in the worddocument	a. Exploring various symbols available b. Inserting a symbol in the text c. Inserting mathematical equations in the document
13.	Get familiarized with MS Excel Ribbon layout	a. Get familiarized with excel layout b. Use various features available in toolbar	a. Open /create an MS Excel spread sheet and familiarity with MS Excel layout b. Use Quick Access Toolbar, Title Bar, Worksheets, Formula Bar, Status Bar
14.	Access and Enter data in the cells	a. Access and select the required cells by various addressing methods b. Enter and edit data	a. Moving around a Worksheets using Quick access toolbar b. Selecting Cells, Entering Data-Editing a Cell, Wrapping of Text-Deleting a Cell Entry, Saving a File, Closing Excel
15.	Edit spread sheetselect, Copy, Cut, Paste	Format the excel sheet	a. Inserting and Deleting Columns and Rows b. Creating Borders c. Merging and aligning center d. Adding background Color Changing the Font, Font Size, and Font Color e. Formatting text with Bold, Italicize, and Underline f. Working with Long Text, Change a Column's Width
16.	Use built in functions and Format Data	Use built in functions in Excel	a. Performing Mathematical Calculations b. Verification AutoSum c. Perform Automatic Calculations d. Aligning Cell Entries
17.	Create Excel Functions, Fill Cells	a. To Create Excel sheets involving cross references and equations b. Using the advanced functions for conditional calculations	a. Using Reference Operators b. Working with sum, Sum if , Count and CountIf Functions c. Filling Cells Automatically
18.	Enter a Formula for automatic calculations	Enter formula for automatic calculations	a. Entering formulae b. Using Cell References in Formulae c. Using Automatic updating function of Excel Formulae d. Using Mathematical Operatorsin Formulae e. Using Excel Error Message andHelp
19.	Sort and filter data in sheet.	a. Refine the data in a worksheet and keep it organized. b. Narrow a worksheet by selecting specific choice	a. Sorting data in multiple columns b. Sorting data in a row c. Sorting data using Custom order Filter data in work sheet

20.	Practice Excel Graphs and Charts	a. Use data in Excel sheet to Create technical charts and graphs b. Prepare various graphs from data.	a. Using data in sheets for getting charts. b. Producing various charts.
21.	Format a Worksheet in Excel, use page setup andprint features	Format Excel sheet, Insert headers & footers and print	a. Shading alternate rows of data b. Adding currency and percentage symbols c. Changing height of a row andwidth of a column d. Changing data alignment e. Inserting Headers and Footers f. Set Print Options and Printing.
22.	Develop lab report formats of respective discipline	Use Headers/Footers/Page Numbers for preparing reports	Creating Lab reports using MSExcel
23.	Get familiarized with Ribbon layout &features of PowerPoint.	Access required options in the tool bar	Explore and use various options inPowerPoint a. Home b. Insert c. Design d. Animation e. Slideshow f. View g.Review
24.	Create a simple PowerPoint Presentation	a. Create simple PowerPoint presentation with photographs/ClipArt and text boxes b. Use bullets option	a. Inserting a New Slide into PowerPoint b. Changing the Title of a PowerPoint Slide c. Using Bullets in PowerPoint d. Adding an Image to aPowerPoint Slide e. Adding a Textbox to a PowerPoint slide
25.	Set up a Master Slide in PowerPoint and add notes	a. Setup Master slide and format b. Add notes to master slide.	a. Creating a PowerPoint Design Template b. Modifying themes c. Switching between Slide master view and Normal view d. Formatting a Design Template for Master Slide e. Adding a Title Slide to a Design Template f. Using the Slide Show g. Adding Notes to a PowerPoint Presentation slide

26.	Insert Text andObjects	a. Insert Text and Objects. b. Use 3d features	a. Inserting Text and objects b. Setting Indents and line spacing c. Inserting pictures/ clipart d. Formatting pictures e. Inserting shapes and word art f. Using 3d features to Arrange objects
27.	Create Flow Charts / Organizational Charts	Create organizational charts and flow charts using smart art	a. Creating a Flow Chart in PowerPoint b. Grouping and Ungrouping Shapes c. Use smart art
28.	Insert Tables	Insert tables and format	a. Using Tables in PowerPoint b. Formatting the Table Data c. Changing Table Background
29.	Insert Charts/Graphs	Create charts and Bar graphs, Pie Charts and format.	a. Creating 3D Bar Graphs in PowerPoint b. Working with the PowerPoint Datasheet c. Formatting a PowerPoint ChartAxis d. Formatting the Bars of a Chart e. Creating PowerPoint Pie Charts f. Using Pie Chart Segments g. Creating 2D Bar Charts in PowerPoint h. Formatting the 2D Chart i. Formatting a Chart Background
30.	Insert audio & video, Hyperlinks in a slide and Add narration to the slide	a. Insert Sounds and Video in appropriate format. b. Add narration to the slide. c. Use hyperlinks to switch to different slides and files	a. Inserting sounds in the slide and hide the audio symbol b. Adjusting the volume in the settings c. Inserting video file in the format supported by PowerPoint ina slide d. Using automatic and on click options e. Adding narration to the slide Insert Hyperlinks
31.	Create Animationeffects	Add animation effects	a. Applying transitions to slides b. Using special animation effects like Entrance, Emphasis, Motion Paths & Exit as per requirement.
32.	Reviewing presentation	a. Use Spell and Grammar check feature b. Setup slide show c. Add timing to theslides d. Setup automatic slide show	a. Checking spelling and grammar b. Previewing presentation c. Setting up slide show d. Setting up resolution e. Using RehearseTiming feature in PowerPoint f. Using PowerPoint Pen Tool during slide show g. Saving h. Printing presentation Slides as Hand- out
33	Familiarizing with AI Tools	Introductions of AI tools and their applications. Understand the basic use cases and functionality of AI tools (like	a) Grasping the concept of Artificial Intelligence and how tools mimic human thinking or behavior. b) Identifying and interacting with AI tools

		ChatGPT, Google Gemini, Teachable Machine, etc.).	such as: **ChatGPT** (natural language processing), **Google Teachable Machine** (image/audio classification), **DALL·E / Bing Image Creator** (AI art), **Grammarly / Quillbot** (AI-based writing assistants).
34	Usage of ChatGPT	a) Introduction to ChatGPT, an AI-powered conversational assistant. b) To explore ChatGPT's capabilities in answering questions, generating content, and solving problems.	a) Operating the ChatGPT interface (web or app), input prompts, and interpret outputs. b) Using ChatGPT to generate summaries, ideas, code snippets, explanations, emails, etc. c) Evaluating the relevance and accuracy of ChatGPT's responses.
35	Object identification using AI Tools based on CNN,YOLO,SSD,R-CNN	a. Get awareness about object detection techniques using AI. b. To explore how AI tools based on **CNN, YOLO, SSD, and R-CNN** detect and classify objects in images/videos.	a) Differentiating object detection from image classification. b) Using web-based AI tools or platforms that demonstrate object detection (e.g., Teachable Machine, Roboflow, Edge Impulse, Hugging Face Demos). c) Observing and comparing the speed, accuracy, and bounding box behavior of different models.
36	Paraphrase text using AI Tools(PEGASUS,GPT,T5)	a. Get awareness about AI-powered text paraphrasing techniques. b. To explore the usage and functioning of transformer-based models like **PEGASUS**, **GPT**, and **T5**	a) Recognizing of Natural Language Processing (NLP) tasks and how transformer models like PEGASUS, GPT, and T5 can be used. b) Using AI tools to generate reworded versions of sentences or paragraphs while retaining the original meaning. c) Interacting with user-friendly interfaces like: • Hugging Face demos • ChatGPT • Quillbot • Parrot.ai
37	Text-to-Image Generation using AI Tools (DALL-E,MIDJOURNEY)	a) Get awareness about **text-to-image generation** using advanced AI models. b) To explore the usage of tools like **DALL·E** and **Mid journey** convert text prompts into realistic or artistic images.	a. Learning usage of how AI models generate visual content from natural language prompts. b. Formulating effective, clear, and creative text prompts to generate meaningful images. c. Enhancing creative thinking by translating ideas into visual representations using AI.

			d. Analyzing and comparing output quality, style, and relevance between DALL·E and Midjourney.
38	Voice Command Simulation using AI Tools (SPEECH-TO-TEXT)	Get awareness about **Speech-to-Text (STT)** technology and its role in AI-powered voice recognition systems.	a. Using AI tools to generate text from speech. b. Reading prompts and commands to analyze how accurately the tool transcribes voice. c. Using voice to simulate commands such as opening files, dictating emails, or interacting with virtual assistants.
39	Usage of ML Tools	a) To use ML tools for suitable real-world applications b) To use **popular ML tools and platforms** through simple, hands-on demonstrations.	a. Understanding key ML terms like dataset, training, testing, classification, prediction, and accuracy. b. Learning to use beginner-friendly ML tools such as: • **Teachable Machine by Google** (image/audio recognition) • **Microsoft Lobe** (no-code image classification) • **Weka** (GUI-based ML toolkit) • **IBM Watson Studio** (visual data workflows)
40	Usage of Quantum Computing Tools	To explore and interact with **quantum computing simulation tools** and platforms.	a. Understanding key terms: **Qubit**, **Superposition**, **Entanglement**, **Quantum Gate**, **Quantum Circuit**. b. Navigate and use beginner-friendly quantum computing tools: • **IBM Quantum Experience (IBM Q / Qiskit)** • **Microsoft Quantum Development Kit** • **Quirk (online quantum circuit simulator)** • **Quantum Playground by Google**
41	To familiarize with quantum bits (qubits) using Dirac notation	a) To introduce the concept of a **qubit** as the fundamental unit of quantum information. b) To understand the representation of qubits using **Dirac (bra-ket) notation**.	• Identify and interpret the basic qubit states: • $\|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}$ • $\|1\rangle = \begin{bmatrix}0\\1\end{bmatrix}$ • Understand that a qubit can exist in a **superposition**: a) Learn how to write and read quantum states using the ket ($\|\rangle$) and bra ($\langle\|$) notations. b) Understand the purpose of $\langle\psi\|$ and how it represents a dual vector in quantum mechanics.

42	To familiarize the behavior of single and multiple qubit gates.	To understand the concept of **quantum gates** and their role in quantum circuits.	a) Recognize the function and matrix representation of:
			
			• **Single-qubit gates**:
			  ○ **Pauli-X (NOT)**: flips $\|0\rangle \leftrightarrow \|1\rangle$
			  ○ **Hadamard (H)**: creates superposition
			  ○ **Pauli-Z**: applies a phase flip
			• **Multi-qubit gates**:
			  ○ **CNOT**: flips target qubit based on control
			  ○ **Toffoli (CCNOT)**: controlled-controlled NOT
			  ○ **SWAP**: exchanges the states of two qubits

43	To familiarize with Qubit as a Coin / Spin Analogy	a) To introduce the concept of a **qubit** using intuitive physical analogies. b) To help students understand **quantum superposition** through the **coin toss** or **spin-½ particle** analogy.	a. Relate a **qubit in superposition** to a **coin spinning in the air**: • Classical coin: heads (0) or tails (1) • Spinning coin: both until observed ($\|0\rangle$ and $\|1\rangle$ at once) b. Use **spin analogy**: a particle with spin "up" ($\|0\rangle$) or "down" ($\|1\rangle$), or in between (superposition)

Table specifying the scope of syllabus to be covered for unit tests

Unit Test	Learning outcomes to be covered
Unit test-1	From 1 to 12
Unit test-2	From 13 to 32
Unit test-3	From 33 to 43

THIRD SEMESTER								
THEORY SUBJECTS	COMPULSORY SUBJECTS	**S No**	**Sub. Code**	**Subject**	**Practicum**	**Periods**	**Total Periods**	**Credits**
		1	26CM301T	DATA STRUCTURES THROUGH PYTHON	N	6	90	4
		2	26CM302T	DATABASE MANAGEMENT SYSTEMS	N	6	90	4
		3	26CM303T	OPERATING SYSTEMS	N	6	90	4
	ELECTIVE SUBJECTS		ELECTIVE SUBJECT					
		4	26CM304E	ENGINEERING MATHEMATICS II	N	3	45	2
		5	26CM305E	DIGITAL ELECTRONICS				
		6	26CM306E	COMPUTER ORGANIZATION				
	AUDIT SUBJECT							
		7	26CM307A	MULTIMEDIA	Y	2	30	
			TOTAL THEORY + PRACTICUM PERIODS			**23**	**345**	**14**
PRACTICAL SUBJECTS	COMPULSORY SUBJECTS	8	26CM308L	DATA STRUCTURES THROUGH PYTHON LAB	N	6	90	2
		9	26CM309L	DBMS LAB	N	6	90	2
		10	26CM310L	LINUX PRACTICALS	Y	4	60	1.5
				ACTIVITY	…	3	45	0.5
			TOTAL PRACTICAL + PRACTICUM PERIODS			**19**	**285**	**6**
GRAND TOTAL PERIODS (15 WEEKS PER SEMESTER)						**42**	**630**	**20**

DATA STRUCTURES THROUGH PYTHON

TIME SCHEDULE:

Course code	Course Title	No. of Periods/Weeks	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM301T	Data Structures Through Python	6	90	30	70	4

S.No.	Chapter/Unit Title	No.of Periods	Weightage of Marks	No.of Short Questions	No.of Essay Questions	CO's Mapped
1.	Introduction to Object Oriented Programming(OOP)	15	14	2	1	CO1
2.	Introduction to Data Structures	20	22	2	2	CO2
3.	Linear Data structure-Linked lists	20	22	2	2	CO3
4.	Linear Data structure –Stacks	15	14	2	1	CO4
5.	Queues and Trees	20	22	2	2	CO5
	Total Periods	90	94	10	8	

COURSE OBJECTIVES:

At the end of the course students will be able to

i)To know the fundamentals of OOP

iI)To know the various types of Data Structures

iiI)To familiarize with the representation of Data Structures

iv)To use various Data structures in organizing data

v)To reinforce theoretical concepts by writing relevant programs

COURSE OUTCOME:

CO1	CM301.1	Know about Object oriented programming concepts
CO2	CM301.2	Illustrate various techniques of sorting and searching
CO3	CM301.3	Explain the operations on Various Linked Lists
CO4	CM301.4	Apply the operations of Stack.
CO5	CM301.5	Explain the operations of Queues and Apply Binary tree traversal techniques.

LEARNING OUTCOMES:

1.Introduction to Object Oriented Programming(OOP)

1.1 Define OOP

1.2 Differentiate between procedural oriented programming (POP) and object-oriented programming (OOP).

1.3 List and Explain the core principles of OOP.

1.4 State the importance of Data Abstraction and Encapsulation

1.5 Describe Inheritance and Polymorphism.

1.6 Define Class and Object in Python.

1.7 Know the creation of classes in Python

1.8 Know the creation of objects (instances) in Python

1.9 State the importance of __init__ method (constructor) and self.

1.10 Explain Method Overloading .

1.11 Explian Method Overriding .

1.12 Explain various types of inheritance.

2. Introduction to Data Structures

2.1 Define Data structure

2.2 List various types of Data Structures

2.3 Explain Linear Data Structures

2.4 Explain Non-Linear Data Structures

2.5 Know about Time and Space complexities

2.6 List and Explain Abstract Data Types

2.7 List and Explain Built-in data structures.

2.8 Various Sorting Techniques

2.8.1 Define Sorting

2.8.2 List the methods of Sorting

2.8.3 Explain Bubble Sort .

2.8.4 Explain Selection Sort.

2.8.5 Explain Insertion Sort

2.8.6 Explain Quick Sort

2.8.7 Explain Merge Sort

2.9 Various Searching Techniques

- 2.9.1 Define searching
- 2.9.2 List types of searching techniques
- 2.9.3 Explain Linear Search with Recursion and with-out Recursion
- 2.9.4 Explain Binary Search with Recursion and with-out Recursion

3. Linear Data structure : Linked lists

3.1 Explain arrays in Python

3.2 Explain the following for Singly Linked List

- 3.2.1 Creation
- 3.2.2 Perform insertion, deletion and traverse operations
- 3.2.3 Perform search and replace an element

3.3 Explain the following for Doubly Linked List

- 3.3.1 Creation
- 3.3.2 Perform insertion, deletion, and traverse operations
- 3.3.3 Perform search and replace an element

3.4 Explain the following for Singly Circular List

- 3.4.1 Creation
- 3.4.2 Perform insertion, deletion, and traverse operations
- 3.4.3 Perform search and replace an element

3.5 Explain the following for Doubly Circular List

- 3.5.1 Creation
- 3.5.2 Perform insertion, deletion and traverse operations
- 3.5.3 Perform search and replace an element

3.6 Differentiate between arrays and linked lists.

4. Linear Data Structures-Stacks

4.1 Define Stack

4.2 Explain the push, pop and display operations of a Stack

4.2 Explain implementation of a Stack using built-in functions in python

4.3 Explain implementation of a Stack without using built-in functions.

4.4 Explain Linked List implementation of a Stack with various operations.

4.5 List the applications of Stacks

4.6 Convert Infix expression to Postfix expression

4.7 Explain the steps for Conversion of Infix expression to Postfix expression with program

4.8 Explain the steps for Evaluating Postfix expression with program

4.9 Evaluate Postfix expression

5. Queues and Trees

5.1 Define Queue

5.2 Explain the insertion, deletion and display operations on Queues

5.3 Explain the implementation of a Queue using built-in functions in python

5.4 Explain the implementation of a Queue without using built-in functions.

5.5 Explain Linked List implementation of a Queue with various operations..

5.6 Explain the implementation of a Circular Queue.

5.7 List the application of Queues

5.8 Non Linear Data Structures-Trees:

5.8.1 Define a Tree
5.8.2 Know about Root, Edge, Parent, Child, Siblings, Leaf, Internal nodes,
Degree, Level, Height, Depth, Path, Sub tree, Forest.
5.8.3 Define Binary Tree
5.8.4 Differentiate between General Tree and Binary Tree.
5.8.5 Define Binary Search Tree
5.8.6 Perform various traversals on Binary Search Trees
5.8.7 Construct a Binary Tree using In-order and Preorder Traversals
5.8.8. Construct a Binary Tree using In-order and Post-order Traversals
5.8.8 Perform insertion, deletion, search and various traversal operations
on a Binary Search Tree.

CO-PO/PSO MATRIX:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM301.1	**2**	**3**	**1**	**2**	**2**	**1**	**1**	**2**	**1**	**3**
CM301.2	**1**	**2**	**1**	**2**	**2**			**2**	**2**	**2**
CM301.3	**2**	**3**	**1**	**2**	**3**			**2**	**1**	**2**
CM301.4	**2**	**1**	**1**	**1**	**1**			**2**	**3**	**1**
CM301.5	**1**	**2**	**1**	**2**	**2**	**1**	**1**	**2**	**2**	**3**
Average	**2**	**2**	**1**	**2**	**2**	**1**	**1**	**2**	**2**	**2**

3=Strongly mapped , 2=moderately mapped, 1=slightly mapped

COURSE CONTENT:

1. Concepts of Object Oriented Programming:

Introduction to OOP – difference between procedural oriented programming (POP) and object oriented programming (OOP) – core principles of OOP – Abstraction and Encapsulation–inheritance and Polymorphism – class and object in Python - creation of classes in Python – creation of objects in Python – __init__ method (constructor) and self – Method Overloading – Method Overriding – various types of inheritance.

2. Introduction to Data structures

Data structure – types of Data Structures – Time and Space complexities – Data Types and Abstract Data Types – Built-in data structures such as Lists, Tuples, Dictionaries, Sets – sorting techniques – Bubble sort , Selection sort, Insertion sort, Quick sort, Merge sort – Searching Techniques: linear and Binary

3.Linear data structures-Linked Lists

Linked Lists – Types - Singly Linked Lists – Create, insert, delete, sort, search and replace an element in a linked list – Doubly Linked Lists – Create, insert, delete, sort, search and replace an element in a linked list – Define terms Singly circular and doubly circular linked lists

4. Linear data structures-Stacks

Stacks- Implementation of stacks, application of stacks, converting infix to postfix expression and postfix expression evaluation.

5. Queues and Trees

Queues–Implementation of queues- Application of queues- know about Circular queues, and Priority queue.

Trees- Trees- Trees Terminology–Binary trees –Representation – Binary Search Tree-various operations-Tree traversals

REFERENCE BOOKS

1. "Data Structures and Algorithms in Python" by Michael T. Goodrich, Roberto Tamassia, and Michael H. Goldwasser
2. "Data Structures and Algorithmic Thinking with Python" by Narasimha Karumanchi
3. "Hands-On Data Structures and Algorithms with Python" by Dr. Basant Agarwal
4. https://www.geeksforgeeks.org/dsa/python-data-structures-and-algorithms/
5. https://www.w3schools.com/python/python_dsa.asp

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TESTS

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 3.3
Unit test-2	From 3.4 to 5.8.8

Course code	Course Title	No. of Periods/Weeks	Total No. of periods	Marks for FA	Marks for SA	Credits
C26CM302T	**DBMS**	**6**	**90**	**30**	**70**	

TIME SCHEDULE:

S.No.	Chapter/Unit Title	No.of Periods	Weightage of Marks	No.Of Short Questions	No.Of Essay Questions	CO's Mapped
1.	Concepts of DBMS & RDBMS	20	22	2	2	CO1
2.	Concepts of SQL	20	22	2	2	CO2
3.	Basics of PL/ SQL	20	22	2	2	CO3
4.	Advance PL/SQL	15	14	2	1	CO4
5.	Concepts of NoSQL & MongoDB.	15	14	2	1	CO5
Total Periods		90	94	10	08	

COURSE OBJECTIVES:

i)To know the fundamentals of DBMS

ii)To familiarize insert, retrieve, update, delete data in database

iii)To familiarize programming skills for insert, retrieve, update, delete data in database

COURSE OUTCOME:

CO1	CM302.1	Describe fundamentals, types and Overall structure of DBMS
CO2	CM302.2	Apply SQL commands to create, retrieve, update, delete data from the Relational data bases.
CO3	CM302.3	Describe PL/SQL programming constructs, control statements and sub programs.
CO4	CM302.4	Apply cursors, triggers and Exception handling concepts

CO5	CM302.5	Use NOSQL database concepts and MongoDB data base concepts in designing database Schema.

LEARNING OUTCOMES:

1.0 Concepts of DBMS &RDBMS

1.1 Define Database Management System (DBMS)
1.2 List the advantages of DBMS
1.3 Explain Database Abstraction, Data Independence
1.4 Define Instance and Schema
1.5 Explain Data Models.
1.6 Define Database languages DDL, DML, TCL
1.7 Explain Database Administrator, Users and Database System Architecture with diagram.
1.8 Define Entity, Entity sets, Relationship, Relationship sets, Super Key , Candidate Key and Primary Key, Foreign Key
1.9 Explain Mapping Cardinalities.
1.10 List the symbols used in ER model.
1.11 Know The Entity-Relationship Model.
1.12 Reduce the ER-diagrams to tables
1.13 Explain Generalization, Specialization & Aggregation.
1.14 Explain Functional Dependencies, Normalizations– 1 NF, 2 NF and 3NF

2.0 Concepts of SQL

2.1 Explain SQL and benefits of SQL.
2.2 Explain literals & different data types like character, number, long, date, raw and long raw etc.
2.3 Illustrate the comments within SQL Statement
2.4 Explain SQL Operators
2.5 Describe Data Definition Language commands CREATE, ALTER and DROP.
2.6 Explain integrity constraints through creating a table and altering table.
2.7 Describe Data Manipulation Language commands INSERT, UPDATE and DELETE
2.8 Explain SELECT statement with WHERE, ORDER BY, GROUP BY and HAVING clauses with examples
2.9 List and explain single row(Number, character, date and conversion) functions
2.10 List and Explain group functions
2.11 Explain Transaction Control Commands COMMIT, SAVEPOINT, ROLLBACK, GRANT, and REVOKE.
2.12 Explain Subqueries with examples
2.13 Explain Joins (Equi Join, Non-Equi Joins, Inner Join, Outer Join, cross join and Self join) with syntax and examples.

3.0 PL/SQL

3.1 Explain PL/SQL Block structure.
3.2 List the features of PL/SQL
3.3 Explain the data types of PL/SQL
3.4 Declaration of variables
3.5 Explain PL/SQL tables and user defined records.
3.6 Explain Input/Output statements

3.7 Explain decision making statements and illustrate
3.8 Explain looping statements and illustrate
3.9 Define procedure and function
3.10 Describe the advantages of subprograms.
3.11 Explain handling procedures and functions with example programs.
3.12 Explain the parameter modes in PL/SQL with examples (in , out and in out)

4.0 Advanced PL/SQL and Exception Handling

4.1 Define cursor.
4.2 Classify cursors
4.3 Explain implicit cursor with example
4.4 Explain explicit cursors with example
4.5 Define trigger
4.6 List Advantages of triggers
4.7 Explain database triggers.
4.8 Define the term Exception handling
4.9 List the advantages of Exception handling
4.10 List various types of Exception handling
4.11 Illustrate built-in Exceptions
4.12 Illustrate User defined Exceptions

5.0 Concepts of NoSQL & MongoDB.

5.1 No SQL
- 5.1.1 List features of NOSQL
- 5.1.2 Compare RDBMS and NoSQL
- 5.1.3 List the Advantages and Disadvantages of NoSQL
- 5.1.4 Know about the ACID and BASE system.
- 5.1.5 Compare ACID and BASE properties
- 5.1.6 NoSQL
  - 5.1.6.1 Key-value stores,
  - 5.1.6.2 Column-oriented,
  - 5.1.6.3 Graph oriented Databases
  - 5.1.6.4 Document oriented Databases.

5.2 MongoDB
- 5.2.1 What is MongoDB
- 5.2.2 List the advantages of MongoDB
- 5.2.3 Explain the Creation, Dropping of Collection in MongoDB
- 5.2.4 Explain the creation and Dropping of Database in MongoDB
- 5.2.5 Explain the Datatypes of MongoDB.
- 5.2.6 Explain Inserting Document, Query Document, Update Document, Deleting Document & Sorting Document.

CO-PO/PSO MATRIX:

CO NO.	PO1	PO2	PO 3	PO4	PO5	PO6	PO7	PSO 1	PSO 2	PSO 3
CM302.1	**3**	**2**	**2**	**2**	**2**	**3**	**2**	**2**	**3**	**1**
CM302.2	**2**	**3**	**3**	**3**	**3**	**3**	**3**	**2**	**1**	**2**
CM302.3	**3**	**2**	**2**	**2**	**1**	**3**	**2**	**2**	**3**	**1**
CM302.4	**1**	**3**	**2**	**2**	**3**	**2**	**3**		**3**	**3**
CM302.5	**3**	**1**	**3**	**1**	**3**	**3**	**2**	**2**	**2**	**3**
Average	**2.4**	**2.2**	**2.6**	**2**	**2.2**	**3**	**2.8**	**2**	**2**	**2**

3=Strongly mapped , 2=moderately mapped, 1=slightly mapped

COURSE CONTENT:

1. Concepts of DBMS & RDBMS

Define DBMS –Advantages of DBMS - Data Abstraction – Data Models – Instance and Schema – Data Independence – Data Definition Language - Data Manipulation Language – Database Administrator - Database Users – Database system Structure.

Entities – Relationships and Relationship sets – Mapping constraints – Entity – Relationship Diagram – Super key , Candidate key and Primary key - Reducing E- R Diagrams to tables – Generalization and Specialization – Aggregation – Functional Dependencies - Normal forms 1NF , 2 NF , 3 NF

2. Concepts of SQL

Benefits of SQL Literals – Text –Integer – Number – Data types – Character data types – Number data type – Long data type –Raw and Long Raw data types – comments within SQL statements

SQL Operators –DDL Commands – Integrity Constraints – DML Commands - functions – single row functions – numeric functions – character functions – date functions – conversion functions – other functions- Group functions. Transaction control commands-Sub queries - Joins.

3. Basics of PL/SQL

SQL features –SQL Block architecture – Data types – Declaration - PL/SQL tables – user defined records.

Conditional control- IF statement --Loop Statements sub programs- Advantages of sub programs – procedures – Functions RETURN statement – parameter modes

4. Advanced PL/SQL

Cursors – Implicit cursor – Explicit cursor – Triggers – Advantages - creating trigger – raising trigger – Data base triggers

5. NoSQL& Basics of MongoDB

Introduction to NoSQL - Features – Comparison of RDBMS and NoSQL- Advantages and Disadvantages of NoSQL - BASE system – ACID System – Comparison of ACID and BASE properties – Classification of NoSQL as Key-value stores, Column-oriented, Graph and Document oriented Databases- Introduction to MongoDB - advantages of MongoDB - MongoDB - Installation of MongoDB - Creation of Database - Dropping of Database - Creation of Collection - Dropping of Collection - Data types of MongoDB -- Inserting Document - Query Document - Updating Document – Deleting Documents - Sorting Documents

REFERENCE BOOKS:

1. Database System Concepts -- Silberschatz, Henry F. Korth, S. Sudarshan
2. Oracle Database 11g :The Complete Reference - Kevin Loney
3. Understanding ORACLE -- James T. Peary & Joseph G. Laseer.
4. RDBMS with ORACLE -- Rolland.
5. ORACLE series books of ORACLE Press – TMH.
6. Starting out with Oracle – Covering Databases- John Day &CraigVan
7. PL/SQL, Developer Tools & DBA -- Slyke, Dreamtech
8. www.nosql-database.org
9. www.mongodb.org
10. https://www.guru99.com/nosql-tutorial.html#5

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TESTS

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 3.5
Unit test-2	From 3.6 to 5.2.5

Course code	Course Title	No. of Periods/Weeks	Total No. of periods	Marks for FA	Marks for SA
26CM303T	Operating Systems	6	90	30	70

TIME SCHEDULE

S.No.	Chapter/Unit Title	No.of Periods	Weightage Allocated	No. of Short Answer Questions	No. of Essay Questions	CO's Mapped
1	Introduction to Operating system	15	14	2	1	CO1
2	Process management	20	22	2	2	CO2
3	Synchronization & Deadlocks	20	22	2	2	CO3
4	Memory management	20	22	2	2	CO4
5	Disk scheduling and File management	15	14	2	1	CO5
Total Periods		90	94	10	8	

COURSE OBJECTIVES

Upon completion of the course the student shall be able to	
(i)	To know about the basics of Operating Systems.
(ii)	To familiarize with process management and Scheduling algorithms.
(iii)	To familiarize with Synchronization and deadlock techniques.
(iv)	To understand various Memory management techniques.
(v)	To familiarize with File management.

COURSE OUTCOMES

CO1	CM303.1	Explain basic concepts of Operating System
CO2	CM303.2	Explain process scheduling algorithm
CO3	CM303.3	Describe Semaphores, solving problems using semaphores, synchronization and Deadlock handling techniques
CO4	CM303.4	Use memory management techniques and page replacement algorithms
CO5	CM303.5	Use Disk scheduling algorithms and File allocation methods with respect to different operating systems

LEARNING OUTCOMES

1.0 Introduction to operating systems

1.1 Define an operating system

1.2 Discuss history of operating system

1.3 Discuss about various types of operating systems

1.4 Distinguish spooling and buffering

1.5 Explain the concepts

- 1.5.1 Multiprogramming
- 1.5.2 Timesharing Systems
- 1.5.3 Distributed Systems
- 1.5.4 Real time systems

1.6 Differentiate between distributed and real time systems

1.7 Describe multiprocessor systems

1.8 Describe the operating system components

1.9 Discuss operating system services

1.10 Define system call with an example.

1.11 List and explain different types of system calls

1.12 Define single user, multi user operating system structure

2.0 Process management

2.1 Define process and process control block

2.2 Explain process state diagram

2.3 Describe process creation and termination

2.4 Discuss the relation between processes

2.5 Define Thread and describe multithreading

2.6 Explain scheduling concepts

2.7 Describe scheduling queues and schedulers

2.8 Explain CPU scheduling and scheduling criteria

2.9 Explain various scheduling algorithms

2.9.1 FCFS

2.9.2 SJF

2.9.3 Round Robin

2.9.4 Priority

2.9.5 Multilevel Scheduling

3.0 Synchronization & Deadlocks

3.1 Describe semaphores

3.2 Explain semaphores to solve Producer consumer problem

3.3 Explain semaphores to solve Readers Writers Problem

3.4 Explain inter process communication

3.5 Define Deadlock

3.6 State the necessary conditions for arising deadlocks

3.7 State various techniques for deadlock prevention

3.8 Discuss Deadlock Detection and Avoidance strategies.

3.9 Describe the process of recovering from deadlock

3.10 Explain Dining philosopher's problem

4.0 Memory management

4.1 Describe briefly address binding, dynamic loading, dynamic linking

4.2 Define overlays

4.3 Describe swapping

4.4 Explain single partition allocation

4.5 Explain multiple partition allocation

4.6 Explain the concept of fragmentation

4.7 Explain paging concept

4.8 Explain how logical address is translated into physical address

4.9 Explain segmentation and segmentation with paging

4.10 Define virtual memory techniques

4.11 Explain demand paging

4.12 Describe page replacement

4.13 Discuss page replacement algorithms

4.13.1 FIFO

4.13.2 LRU

4.13.3 Optimal

4.14 Describe the concept of thrashing

4.15 Explain working set model and page fault frequency

5.0 Disk scheduling and File management

5.1 List and explain various disk performance parameters

5.2 Disk scheduling policies

5.2.1 FCFS

5.2.2 SSTF

5.2.3 SCAN

5.3 Define file management

5.4 List and explain various file operations

5.5 List and explain various access methods

5.6 List and explain various allocation methods

5.6 List and explain directory structure

5.7 Explain disk organization and structure

CO-PO/PSO MATRIX

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM303.1	3	2	2	2	3	3	3	3	2	2
CM303.2	3	3	3	2	2	3	3	2	2	2
CM303.3	3	3	3	2	3	3	3	3	2	3
CM303.4	3	3	3	3	3	3	3	3	3	3
CM303.5	3	3	2	2	2	3	3	3	3	3
Average	3	3	2.5	2	2.5	3	3	3	2.5	2.5

3=Strongly mapped , 2=moderately mapped, 1=slightly mapped

COURSE CONTENT

1. Introduction to operating systems

Operating System –Evolution of operating system-Types of Operating Systems - Multi Programming and Time Sharing - Distributed and Real time Systems - spooling and buffering - Multi processor systems-Components of Operating Systems - operating System Services - system Calls - single User and Multi user operating System Structure.

2. Process management

Processes - Sequential Processes - Process State Diagram - Process Control Block - Process Creation and Termination - Relations between Processes - Threads and Multi

Threading - Scheduling Concepts - Schedulers - CPU scheduling and Scheduling criteria - scheduling algorithms.

3. Synchronization & Deadlocks

Semaphores –Application of semaphores-Producer Consumer problem- Readers-writers problem – Inter Process Communications - monitors- Deadlocks – Necessary conditions of deadlock - deadlock prevention - deadlock detection - deadlock avoidance-Deadlock Recovery-Dining philosopher problem.

4. Memory management

Address binding -Dynamic Loading- dynamic linking-overlays-swapping- memory allocation-fragmentation-paging-segmentation- Address Translation-segmentation with paging-Benefits of virtual memory - virtual memory techniques - demand paging - page replacements - page replacement algorithms – thrashing.

5.Disk scheduling and File management

Disk performance parameters - Disk scheduling policies – Introduction to file systems - File management-File Operations - Access methods - Directory structure organization - File Protection.

REFERENCE BOOKS

1. Operating Systems -- Silber Schatz and Galvin
2. Operating Systems -- William Stallings, PHI
3. Operating Systems -- Dietel and Dietel
4. Operating Systems -- Dhamdhere (TMH)
5. Advanced Operating Systems -- Tanenbaum

Table specifying the scope of syllabus to be covered for unit tests

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 3.3
Unit test-2	From 3.4 to 5.7

C-26 Open Elective-XXX

ENGINEERING MATHEMATICS-II

(Common to all Branches)

Course Code	Course Title	No. of Periods/week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM304E	Engineering Mathematics-II	3	45	30	70	3

S.No.	Unit Title	No. of periods	COs mapped
1	Applications of Definite Integrals.	15	CO1
2	Differential Equations	15	CO2
3	Probability & Statistics	15	CO3
	Total Periods	**45**	

TIME SCHEDULE

S.No.	Chapter/Unit title	No. of Periods	Marks Allotted	No. of Short Questions	No. of Essay Questions	COs mapped
Unit – I: Applications of Definite Integrals						
1	Area of curves	4	7	1	½	CO1
2	Volumes of Solids of Revolution	3	4	0	½	CO1
3	Mean and RMS values	4	11	1	1	CO1
4	Numerical Integration	4	8	0	1	CO1
Unit – II: Differential Equations						
5	Introduction to Differential Equations	4	6	2	0	CO2
6	Solution of first order differential equations	6	14	2	1	CO2
7	Solution of second order homogeneous and non-homogeneous linear differential equations	5	11	1	1	CO2
Unit – III: Probability and Statistics						
8	Probability	5	11	1	1	CO3
9	Measures of Dispersion	6	14	2	1	CO3
10	Correlation	4	8	0	1	CO3
	Total	45	94	10	8	
			Marks	30	64	

COURSE OBJECTIVES

(i)	To apply integral techniques to solve various engineering problems.
(ii)	To solve first-order and first-degree differential equations and second-order homogeneous and non-homogeneous linear differential equations.
(iii)	To analyse data using the concepts of probability and statistical techniques.

COURSE OUTCOMES

CO1	Apply definite integrals in engineering applications.
CO2	Solve first-order and first-degree differential equations and second-order homogeneous and non-homogeneous linear differential equations.
CO3	Apply various probability and statistical techniques for data analysis.

LEARNING OUTCOMES

Unit-I

C.O.1 Apply definite integrals in engineering applications.

L.O.1.1 Find the area bounded by a curve and axes.
1.2 Determine the volumes of solids of revolution along the x-axis.
1.3 Obtain the Mean and R.M.S values of simple functions.
1.4 Solve the problems of areas using Numerical Integration.

Unit-II

C.O.2 Solve first-order and first- degree differential equations and second-order homogeneous and non-homogeneous linear differential equations.

L.O.2.1 Define a Differential equation, its order and degree.
2.2 Find order and degree of a given differential equation.
2.3 Form a differential equation by eliminating arbitrary constants.
2.4 Solve the first order and first degree differential equations by variables separable method.
2.5 Solve linear differential equation of the form $\frac{dy}{dx} + Py = Q$, where P and Q are functions of x only or constants.

Syllabus for Unit test-I completed

2.6 Solve Differential equations of the type $(aD^2 + bD + c)\, y = 0$ where $a (\neq 0)$, b, and c are real numbers.
2.7 Define complementary function, particular integral and general solution of a non-homogeneous linear differential equation of second order with constant coefficients.
2.8 Describe the method of solving $f(D)y = e^{ax}$, where f(D) is a polynomial of second order.

Unit-III

C.O. 3 Apply various probability and statistical techniques for data analysis.

L.O.3.1 Recall the basic probability principles.
3.2 State addition theorem of probability for two mutually exclusive and exhaustive events.

3.3 Solve simple problems on addition theorem.
3.4 Explain conditional event and conditional probability.
3.5 Solve simple problems on conditional probability.
3.6 Explain dependent, independent events and state multiplication theorem.
3.7 Solve simple problems on multiplication theorem.
3.8 Recall the measures of central tendency.
3.9 Explain the significance of measures of dispersion to determine the degree of heterogeneity of the data.
3.10 Find the measures of dispersion, Range, Mean Deviation and Standard Deviation for ungrouped data.
3.11 Explain the merits and demerits of these measures of dispersion.
3.12 Explain bivariate data.
3.13 Explain the concept of covariance and correlation between two variables.
3.14 Find Spearman's rank correlation coefficient.

Syllabus for Unit test-II completed

CO/PO – MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1	3	3	3	3				3	3	1
CO2	3	1	1	1				3	1	1
CO3	3	3	3	3				3	3	3
Avg.	3	2.33	2.33	2.33				3	2.33	1.66

Note: The gaps in CO/PO mapping can be met with appropriate activities as follows:

For PO5: Appropriate quiz programmes may be conducted at intervals and duration as decided by concerned faculty.
For PO6: Seminars on applications of mathematics in various engineering disciplines are to be planned and conducted
For PO7: Plan activities in such a way that students can visit the Library to refer standard books on Mathematics and access the latest updates in reputed national and international journals. Additionally, encourage them to attend seminars and learn mathematical software tools.

COURSE CONTENTS

Unit-I: Applications of Definite Integrals

Area bounded by a curve and axes. Volume of Solids of Revolutions. Mean and RMS values of a function on a given interval. Numerical Integration.

Unit -II: Differential Equations

Definition of a differential equation, Order and degree of a differential equation, Formation of differential equations. Solutions of differential equations of first order and first degree using variables separable method and linear differential equation of the type $\frac{dy}{dx} + \mathrm{P}y = \mathrm{Q}$. Solutions of homogenous and non-homogeneous linear differential equations of second order with constant coefficients.

Unit III: Probability & Statistics

Addition theorem of probability, conditional probability, dependent and independent events with multiplication theorem. Measures of dispersion, range, mean deviation and standard deviation of ungrouped data, merits and demerits. Bivariate data, correlation, Spearman's rank correlation coefficient.

TEXTBOOK

Engineering Mathematics-II, a textbook for second year third semester diploma courses, prepared & prescribed by SBTET, AP.

REFERENCE BOOKS

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers.
2. Schaum's Outlines Differential Equations, Richard Bronson & Gabriel B. Costa
3. Schaum's Outline: Introduction to Probability and Statistics, Seymour Lipschutz & John J. Schiller.
4. M.Vygodsky, Mathematical Handbook: Higher Mathematics, Mir Publishers, Moscow.

SUGGESTED E-LEARNING REFERENCES

1. https://www.khanacademy.org/
2. https://www.wolframalpha.com/
3. https://onlinecourses.nptel.ac.in/
4. http://tutorial.math.lamar.edu/

UNIT TEST SYLLABUS

Unit Test	Learning Outcomes to be Covered
Unit Test-I	From L.O 1.1 to L.O 2.5
Unit Test-II	From L.O 2.6 to L.O 3.14

Course code	Course Title	No. of Periods/ Weeks	Total No of periods	Marks for FA	Marks for SA	Credits
26CM305E	**Digital Electronics**	**3**	**45**	**30**	**70**	**2**

TIME SCHEDULE

S. No.	Chapter/Unit Title	No. of Periods	Weightage of marks	No. of Short Answer Questions	No. of Essay Questions	CO's Mapped
1.	Number systems	10	14	2	1	CO1
2.	Logic Gates ,Boolean Algebr and Basic Combinational	10	22	2	2	CO2
3.	Flip-Flops	10	22	2	2	CO3
4.	Counters and registers	10	22	2	2	CO4
5.	Additional Combinational circuits	5	14	2	1	CO5
	Total Periods	45	94	10	8	

COURSE OBJECTIVES

on completion of the course the student shall be able to	
(i)	i)To acquire the basic knowledge of digital logic levels and apply of knowledge to understand digital logic circuits.
(ii)	ii)To prepare students to perform the analysis and design of various digital electronic circuits.

COURSE OUTCOMES

Upon completion of the course the student shall be able to		
CO1	CM-305.1	Explain the structure of various number systems.
CO2	CM-305.2	Describe fundamental concepts and techniques used in digital electronics, the switching algebra theorems and logicgates and apply them to design logic circuits using K-Maps

CO3	CM-305.3	Explain operation of flip flops
CO4	CM-305.4	Design the counting circuits, Registers using flipflop operations.
CO5	CM-305.5	Explain design and implement various sequential circuits, combinational circuits

LEARNING OUTCOMES

1.0 Number systems

1.1 List the various number systems used in digital Computer.
1.2 Explain Decimal number system
1.3 Explain Binary number system
1.4 Explain octal number system
1.5 Explain Hexadecimal number system
1.6 Convert decimal number to other base conversion.
- 1.6.1 Decimal to Binary
- 1.6.2 Decimal to Octal
- 1.6.3 Decimal to Hexadecimal

1.7 Convert binary number to other base conversion.
- 1.7.1 Binary to Decimal
- 1.7.2 Binary to octal
- 1.7.3 Binary to Hexadecimal

1.8 Convert octal number to other base conversion.
- 1.8.1 Octal to Decimal
- 1.8.2 Octal to Binary
- 1.8.3 Octal to Hexadecimal

1.9 Convert hexadecimal other base conversion.
- 1.9.1 Hexadecimal to Decimal
- 1.9.2 Hexadecimal to Binary
- 1.9.3 Hexadecimal to Octal

1.10 Binary numbers representation.
- 1.10.1 Define Binary numbers representation.
- 1.10.2 List the types of Binary numbers representation.
- 1.10.3 Explain Unsigned binary number representation.
- 1.10.4 Explain Signed binary number representation.

1.11 Binary arithmetic.
- 1.11.1 Illustrate addition of two signed binary numbers.
- 1.11.2 Illustrate subtraction of two signed binary numbers.
- 1.11.3 Illustrate binary multiplication.
- 1.11.4 Illustrate Binary division.

1.12 Binary coded decimal (BCD) coding scheme.
- 1.12.1 Define Binary coded decimal (BCD) coding scheme.
- 1.12.2 List the types of Binary coded decimal (BCD)
- 1.12.3 Draw and explain 8421 code.
- 1.12.4 Draw and explain 2421 code.
- 1.12.5 Draw and explain 8 4-2-1 code.

1.13 Character representation
- 1.13.1 List character representation codes
- 1.13.2 Explain the ASCII codingscheme.
- 1.13.3 Explain the EBCDIC codingscheme.

2.0 Boolean algebra , Logic gates and Basic Combinational Circuits

2.1 Boolean algebra

- 2.1.1 Define Boolean algebra
- 2.1.2 Explain AND, OR, NOT operations with truth tables.
- 2.1.3 Explain the working of EX-OR and EX-NOR operations with truth tables.
- 2.1.4 List the different postulates in Boolean algebra.
- 2.1.5 State De-Morgan's theorems.
- 2.1.6 Prove De-Morgan's theorems using truth tables.
- 2.1.7 Apply De-Morgan's theorems and other postulates of Boolean algebra to simplify the given Boolean expression.
- 2.1.8 Generate Boolean expression for given truth table.
  - 2.1.8.1 Using Sum-Of-Products(SOP) method
  - 2.1.8.2 Using Product-Of-Sums(POS)method
- 2.1.9 Use K – map to simplify Boolean expression (up to 4 variables).
  - 2.1.9.1 Using Two variable K-Map
  - 2.1.9.2 Using Three variable K-Map
  - 2.1.9.3 Using Four variable K-Map

2.2 Logic Gates

- 2.2.1 Define Logic gate
- 2.2.2 List basic gates
- 2.2.3 Define OR gate
- 2.2.4 Explain OR gate with logic symbol and truth table.
- 2.2.5 Define AND gate
- 2.2.6 Explain AND gate with logic symbol and truth table.
- 2.2.7 Define NOT gate
- 2.2.8 Explain NOT gate with logic symbol and truth table.
- 2.2.9 What is universal gate? List universal gates
- 2.2.10 Define NOR gate
- 2.2.11 Explain NOR gate with logic symbol and truth table.
- 2.2.12 Define NAND gate
- 2.2.13 Explain NAND gate with logic symbol and truth table.
- 2.2.14 Define EX-OR and EX-NOR gates
- 2.2.15 Explain the working of EX-OR and EX-NOR gates with truth tables.
- 2.2.16 Implement AND, OR and NOT gates using NAND gates only
- 2.2.17 Implement AND, OR and NOT gates using NOR gate only.

2.3 Basic Combinational Circuits

- 2.3.1 Define the Half Adder.
- 2.3.2 Explain the function of Half Adder.
- 2.3.3 Draw Half-Adder circuit using an EX- OR and an AND gate.

2.3.4 Define the Full Adder.
2.3.5 Explain the function of Full Adder.
2.3.6 Construct Full Adder using two Half-Adder and an OR gate
2.3.7 Define the parallel Adder
2.3.8 Explain the function of parallel Adder using logic symbol.
2.3.9 Draw and explain 4-bit parallel adder using full adders.
2.3.10 Explain the working of a serial adder with a block diagram.
2.3.11 List the advantage and disadvantages of a serial adder
2.3.12 List the advantage and disadvantages of a parallel adder.
2.3.13 Distinguish between serial adder and parallel adder.
2.3.14 Explain the operation of a digital comparator circuit for two 4-bitwords.

3.0 Flip-Flops

3.1 List the details of different logic families**.**
3.2 Define positive and negative logic levels.
3.3 Define Flip flop
3.4 Draw and explain the basic principle of operation of aFlip-flop.
3.5 Define Latch.
3.6 Explain the working of NAND latch circuit with truth table and Timing diagram
3.7 Explain the working of a NOR latch circuit with truth table and Timing diagram
3.8 Explain with block diagram, waveforms and truth tables the working of RS Flip-flop.
3.9 Explain with block diagram, waveforms and truth tables the working of D Flip-flop.
3.10 Explain with block diagram, waveforms and truth tables the working of JK Flip-Flop.
3.11 Distinguish between synchronous and asynchronous inputs ofa flip- flop
3.12 State the need for a Master-Slave Flip-Flop.
3.13 Explain the working of a Master-Slave flip-flop using suitable circuit diagram and truth table.

4.0 Counters and Registers

4.1 Counters

4.1.1 Define Counter
4.1.2 List the types of counters.
4.1.3 Define the terms Synchronous counter, Asynchronous counter
4.1.4 Distinguish between asynchronous and synchronous counters.
4.1.5 Draw and explain module-8 ripple counter circuit diagram with waveforms and truth tables
4.1.6 Draw and explain module-10 (decade) Asynchronous counter circuit diagram with waveforms and truth tables
4.1.7 Draw and explain module-8 synchronous counter

circuit diagram with waveforms and truth tables

4.1.8 List the draw backs of ripple counters.
4.1.9 Draw and explain the operation of a 4-bit ring counter.
4.1.10 List the applications of counter.

4.2 Register

4.2.1 Define Register
4.2.2 State the need of Register.
4.2.3 List the methods of data transfer in register.
4.2.4 List the types of Registers
4.2.5 Define Serial in – Serial out register, Serial in – Parallel out register,Parallel in – Serial out register, Parallel in – Parallel out register
4.2.6 Explain the working of serial in – serial out register with circuit diagram.
4.2.7 Explain the working principle of serial in – parallel out register with circuit diagram.
4.2.8 Explain the working principle of shift left Register with circuit diagram.
4.2.9 Explain the working of shift right registers with circuit diagram.

5.0 Additional Combinational Circuits

5.1 Define the terms Multiplexer and Demultiplexer
5.2 Draw and explain the operation of a 4X1 multiplexer circuit diagram with truth table.
5.3 Draw and explain the operation of 1X4 Demultiplexer circuit diagram with truth table.
5.4 List the applications of Multiplexers, Demultiplexers.
5.5 Define the terms Encoder and Decoder.
5.6 Draw and explain the operation of a 4-to-10-linedecoder circuit diagram with truth table.
5.7 Draw and explain the operation of a 8 to 3 encoder circuit diagram with truth table.
5.8 List applications of Decoders, Encoders.

CO-PO/PSO Matrix:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM-305.1	3	3	3		2			3	2	3
CM-305.2	3	3	3	2	1	1		3	2	2
CM-305.3	2	1	1	1			1	1	1	3
CM-305.4	2	2	3	1		2	1	2	2	1
CM-305.5	2	2	3	1	1	1	2	3		1
Average	2.4	2.2	2.6	2.25	1.3	1.3	1.3	2.4	1.75	2

3=Strongly mapped , 2=moderately mapped, 1=slightly mapped

COURSE CONTENTS

1. **Number Systems:** Number Systems, Decimal, Binary, Hexadecimal and Octal codes - Conversion from one number system to another number system - Binary numbers representation -Binary arithmetic - Binary coded decimal (BCD) coding scheme - Character representation codes.

2. Boolean algebra and Logical Gates: Boolean Algebra - AND, OR, NOT ,EX-OR and EX-NOR operations - postulates in Boolean algebra - Boolean theorems, Boolean simplifications of Boolean expressions, Using De-Morgan's theorems- Simplify the given Boolean expression - Generate Boolean expression for given truth table -Using K – map to simplify Boolean expression–
LOGIC GATES – AND,OR,NOT,EX-OR,EX-NOR,NAND and NOR gates - Implement AND, OR, NOT gates using NAND gates only and NOR gates only –
Basic Combinational Circuits – Half adder – Draw Half-Adder circuit using an exclusive OR and an AND gate -Full Adder – Full Adder using two Half-Adder and an OR gate -Parallel Adder – 4-bit parallel adder using full adders - Serial Adder – Advantages and disadvantages of Parallel adder - Advantages and disadvantages of Parallel adder - Differences between serial adder and parallel adder - digital comparator circuit for two 4-bitwords.

3. **FLIP FLOP:** Different logic families - positive and negative logic levels - Basic principles of Flip Flop operation- Explanation of NAND Latch, NOR Latch,RS,D,JK and Master Slave JK flip flops with block diagram, truth table and wave forms.

4. Counters: Basics of Asynchronous, Synchronous counters – Explanation of Asynchronous MOD-8 and MOD-10 counters with diagrams and wave forms- Explanation of Synchronous MOD-8 counter with diagrams and wave forms – 4-bit ripple counter and listing its applications
Registers: Basics of registers –definition, need and types of registers - .definition of SISO,SIPO,PISO and PIPO registers – Explanation of SISO and SIPO registers with diagram- Shift Left and Shift right registers.
5.Additional Combinational Circuits: Definitions of Multiplexer,De-Multiplexers – Explanation of 4X1 Multiplexer – Explanation of 1X4 Demultiplexer – Applications of Multiplexers,Demultiplexers – Definitions of encoder,Decoder - Explanation of 4-to-10-line decoder – Explanation of 8 to 3 encoder – Applications of Encoder,Decoders

REFERENCES

1. Digital principles and applications … Melvino and Leach
2. Digital Electronics…. Bignell – Thomson
3. Modern Digital Electronics…. R.P. JAIN
4. https://www.geeksforgeeks.org/digital-logic/number-system-and-base-conversions/
5. https://www.tutorialspoint.com/digital-electronics/digital-electronics-number-systems.htm

6. https://www.electronics-tutorials.ws/binary/binary-coded-decimal.html
7. https://learnabout-electronics.org/Digital/dig11.php
8. https://www.tutorialspoint.com/digital-electronics/digital-electronics-binary-codes.htm
9. https://www.tutorialspoint.com/digital-electronics/digital-electronics-combinational-circuits.htm
10. https://www.tutorialspoint.com/digital-electronics/digital-electronics-flip-flops.htm
11. https://www.tutorialspoint.com/digital-electronics/digital-electronics-counters.htm
12. https://www.geeksforgeeks.org/digital-logic/counters-in-digital-logic/
13. https://www.tutorialspoint.com/digital-electronics/digital-electronics-shift-registers.htm

UNIT TEST SYLLABUS

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 3.7
Unit test-2	From 3.8 to 5.8

Course code	Course Title	No. of Periods/ Weeks	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM306E	**Computer Organization**	**3**	**45**	**30**	**70**	**2**

TIME SCHEDULE

S.No.	Chapter/Unit Title	No.of Periods	Weightage of marks	NO of short answer Questions	NO of Essay Questions	CO's Mapped
1.	CPU Organization	10	14	2	1	CO1
2.	Information representation and Arithmetic Operations	15	33	3	3	CO2
3.	Memory Organization	10	22	2	2	CO3
4.	I/O Organization	10	25	3	2	CO4
Total Periods		45	94	10	8	

COURSE OBJECTIVES

on completion of the course the student shall be able to	
(i)	To know about Processor organization
(ii)	To familiarize in formation Representation and arithmetic operations
(iii)	To understand how memory and i/o is organized in an effective way

COURSE OUTCOMES

At the end of the course the student able to learn following:		
CO1	CM-306.1	Explain the Basic computer organization techniques
CO2	CM-306.2	Explain various information representations, algorithms of arithmetic operations
CO3	CM-306.3	Explain the memory organization
CO4	CM-304.4	Explain the peripheral organization

LEARNNG OUTCOMES

1.0 CPU ORGANIZATION

1.1 Draw the functional block diagram of Digital computer and explain the function of each unit.

1.2 Define Register

1.3 State the purpose of

- 1.3.1 Accumulator
- 1.3.2 Program counter
- 1.3.3 Instruction Register
- 1.3.4 Memory Buffer Register
- 1.3.5 Memory Address Register

1.4 Draw the block diagram of simple accumulator based CPU.

1.5 Explain the function of each unit.

1.6 Define the terms micro operation, macro operation,

1.7 Define instruction cycle, fetch cycle and execution cycle.

1.8 What is stored program concept

1.9 Explain sequential execution of a program stored in memory by the CPU

2.0 Information representation and Arithmetic Operation

2.1 List and Explain basic types of information representation in a computer.

2.2 Define floating point representation and fixed-point representation of numbers.

2.3 Illustrate the floating point and fixed point representations with example.

2.4 Distinguish between Fixed point and Floating point representations.

2.5 What is Instruction format

2.6 Define terms Opcode, Operand and address.

2.7 Instruction formats with examples

- 2.7.1 Zero address instructions
- 2.7.2 One address instruction
- 2.7.3 Two address instructions
- 2.7.4 Three address instructions

2.8 Define addressing mode

2.9 List and explain various addressing modes.

2.10 Explain the fixed point addition and subtraction operations with flowchart.

2.11 Explain the Fixed point multiplication operation with flowchart.

2.12 Explain the Fixed point division operation with flowchart.

2.13 Explain floating point addition, subtraction operations with flowchart

2.14 Explain floating point multiplication operation with flowchart

2.15 Explain floating division operation with flowchart.

3.0 Memory Organization

3.1 List and state the importance of various types of memories

3.2 Distinguish between main and auxiliary memory.

3.3 State the need for memory hierarchy in a computer.

3.4 Explain memory hierarchy in a computer in detail

3.5 State the significance of various memory device characteristics: access time, access rate, alterability, permanence of storage, cycle time.

3.6 Differentiate between RAM and CAM

3.7 Explain Associative Memory

3.8 Explain the principle of virtual memory organization in a computer system
3.9 Explain virtual address and physical address organization.
3.10 State the principle of locality of reference
3.11 Explain Cache memory organization.
3.12 Explain principle of memory interleaving in a computer.

4.0 I/O Organization

4.1 List peripheral devices that can be connected to a computer.
4.2 State the need for an interface.
4.3 State the importance of bus system.
4.4 List modes of date transfer.
4.5 Explain synchronous and asynchronous data transfer.
4.6 Differentiate between synchronous and asynchronous data transfer.
4.7 Explain hand shaking procedure of data transfer.
4.8 Explain programmed I/O method of data transfer.
4.9 Explain interrupted initiated I/O.
4.10 Explain priority interrupt : polling and daisy chaining priority
4.11 Explain DMA controlled transfer.

CO-PO/PSO Matrix:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM-306.1	3	1	2	1	1	2	1	2	1	1
CM-306.2	3	3	3	2		1		3	2	3
CM-306.3	3	3	3	2		1		3	2	3
CM-306.4	3	2	2	1	2	2	2	2	1	2
Average	3	2.3	2.5	1.5	1.5	1.5	1.5	2.5	1.5	2.3

3=Strongly mapped , 2=moderately mapped, 1=slightly mapped

COURSE CONTENT

1. Processor Organization–Functional block diagram of Digital computer - Definition of Registers – purpose of accumulator, Program counter instruction register, Memory buffer registers, Memory address register - Simple accumulator based CPU and function of each unit. -Stored program concept - Sequential execution of a program stored in memory by the CPU

2. Information representation and Arithmetic Operation- Basic types of information representation - floating point representation and fixed-point representation of numbers, Operand, Opcode and address - Zero address, One address, Two address and Three address instructions - Different addressing modes. -Fixed point addition and subtraction ,FixedPoint multiplication and Fixed division operations with flow charts - floating point addition, subtraction, multiplication and division operations with flowcharts.
3. Memory Organization – Listing of Various types of memories and their importance - Memory hierarchy in a computer -Various memory devices characteristics - Differentiate between RAM and CAM - Associative Memory-

Virtual memory organization in a computer system - Virtual address and physical address organization-Principle of Locality - Cache memory organization – Memory interleaving.

4. I/OOrganization - Peripheral devices - Need for an interface - Bus systems - Modes of date transfer - Synchronous and asynchronous data transfer - Hand shaking procedure of data transfer -Programmed I/O method of data transfer-Interrupted initiated I/O- Priority interrupt: polling and daisy chaining priority- DMA controlled transfer.

Reference Books

1. Structured Computer Organization --- Andrews Tenenbaum.

2. Computer Organization --- Govindarajulu (TMH).

3. Computer Organization & Architecture --- William Stallings

4. Computer System Architecture --- Morris Mano

5.Computer Organization – Car Hamacher, ZvonksVranesic, Safwat Zaky, V Edition, McGraw Hill, 2002.
6.Computer Architecture and Organization – John P. Hayes, Mc Graw Hill International editions, 1998.
7.Computer architecture and organization , 4th edition , P Chakraborty , JAICO publishers.
8. https://pravin-hub-rgb.github.io/BCA/resources/sem3/coatbc304/unit3/index.html
9. https://witscad.com/course/computer-architecture/chapter/data-representation
10. https://medium.com/@shrutiikhatal07/memory-organization-d0f486a4daa0
11.https://pravin-hub-rgb.github.io/BCA/resources/sem3/coatbc304/unit4/index.html

Table specifying the scope of syllabus to be covered for unit tests

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 TO 2.15
Unit test-2	From 3.1 to 4.11

Course Code	Course Title	No. of periods/week	Total No. of periods	Marks for FA	Marks for SA
26CM307A	Multimedia	2	30	NA	NA

TIME SCHEDULE

S.NO	Chapter/ Unit Title	NO. OF PERIODS	CO's mapped
1	**MS Access** - create Database, create table with and without constraints, Insert, delete, update records, implement queries, create relationship between two tables	6	CO1
2	**PageMaker** - Creation of publication using tools, text, shapes, etc, Custom template, colors, text block, Objects, Styles, Page elements, Printing the documents	12	CO2
3	**Photoshop** – Different tools, Working with Layers, Working with painting tools, Colors, Brushes	12	CO3
	TOTAL	30	

COURSE OBJECTIVES

Upon completion of the course the student shall be able to	
(i)	Familiarize with the features of MS Access.
(ii)	Familiarize with the features of Adobe PageMaker
(iii)	Familiarize with the features of Adobe Photoshop

COURSE OUTCOMES

CO1	CM307.1	Working with databases, tables, manipulating records, queries and establishing relationship among tables.
CO2	CM307.2	Create and print publication files such as Textbooks, Visiting Cards, Invitation Cards, etc. using Adobe Pagemaker
CO3	CM307.3	Design and enhance the quality of all types of picture files using Adobe Photoshop

CO-PO/PSO MAPPING MATRIX

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM307.1	3	2	3	1	3	3	3	2	1	2
CM307.2	2	2	3	2	2	2	2	2	2	2
CM307.3	2	3	3	2	1	2	2	2	2	2
Average	2.3	2.3	3	1.6	2	2.3	2.3	2	1.6	2

3=Strongly mapped , 2=moderately mapped, 1=slightly mapped

Learning Outcomes

S.No	Main Learning Outcome	Explanation (Using PPTS, Videos)	Practical
1.	Learn and Practice MS-Access	Explain the concept of databases, tables, operations on tables, constraints, retrieving the data from tables, relating the tables.	1. To create Database To Create table with and without constraints To Insert, delete, update records 2. To implement queries To create relationship between two table
2.	Learn and Practice Adobe Page Maker	Explain various page formatting tools, their usage and various options related to page formatting, Color codes using Adobe page maker	3. Exercise on Installation, invoking and familiarizing Adobe Page Maker. 4. Exercise on Page Maker Tools. 5. Exercise on creating visiting card 6. Exercise on creating book cover page 7. Exercise on creating

			hotel menu card 8. Exercise on creating invitation card 9. Exercise on creating brochure
3.	Learn and Practice Adobe Photoshop	Explain various image editing tools, various operations on images, layers, colors, by using photoshop.	10. Exercise on Installation, invoking and familiarizing Adobe Photoshop 11. Exercise on selection, Resizing & Cropping Images 12. Exercise on Layers 13. Exercise on Photo Retouching 14. Exercise on CREATING SPECIAL EFFECTS 15. Exercise on Exporting Your Work 16. Exercise on Logo Creation

KEY COMPETENCIES

Exp No	Name of the Experiment	Objectives	Key Competencies
1	To create database and create tables with key constraints and insert/delete/update records in a table	a. Open MS Access b. Create database c. save d. create table e. Use primary key f. Insert/delete/update records	❖ Database creation ❖ Table creation ❖ Able to insert/update/delete and delete records in the table
2	To implement queries to extract data from tables Create relationships between tables	a. Open MS Access b. Create database c. Create table d. Use Select command e. Create one more tables f. Insert records g. Use relationship option	❖ Able to display contents of the table based on the user requirement ❖ Able to link tables
3	Exercise on Installation, invoking and familiarizing Adobe Page Maker.	Installation, invoking and familiarizing Adobe Page Maker.	❖ Installing page maker ❖ Explain PageMaker environment
4	Exercise on Page Maker Tools.	Page Maker Tools.	❖ Explain usage of Tool box, zero position, pointer tool, text tool, rotate tool, crop tool, oblique line tool, constrained line tool, box tool, rectangle

			frame, circle tool, circular frame, polygon tool, polygon frame, hand tool and zoom tool.
5	Exercise on creating visiting card	creating visiting card	❖ Create visiting card with text and graphics on both sides with proper formatting.
6	Exercise on creating book cover page	creating book cover page	❖ Create front and back cover page of a book with text and graphics with proper formatting.
7	Exercise on creating hotel menu card	creating hotel menu card	❖ Create a hotel menu card with text and graphics with proper formatting.
8	Exercise on creating invitation card	creating invitation card	❖ Create a invitation card with text and graphics with proper formatting for required no. of pages.
9	Exercise on creating brochure	creating brochure	❖ Create a brochure for the firms like real estate companies, hospitals, educational institutions etc...,
10	Exercise on Installation, invoking and familiarizing Adobe Photoshop	Installation, invoking and familiarizing Adobe Photoshop7.0	❖ Exploring the Toolbox ❖ Exploring Panels & Menus ❖ Creating & Viewing a New Document ❖ Customizing the Interface ❖ Setting Preferences
11	Exercise on Selection, Resizing and Cropping of images	Selection , Resizing and Cropping of images	❖ Understanding various selection methods. ❖ Interpolation Options ❖ Resizing for Print & Web ❖ Cropping & Straightening an Image Adjusting Canvas Size & Canvas Rotation
12	Exercise on LAYERS	Working on Layers	❖ Understanding the Background Layer ❖ Creating, Selecting, Linking & Deleting Layers ❖ Locking & Merging Layers ❖ Copying Layers, Using Perspective & Layer Styles ❖ Filling & Grouping

			Layers ❖ Introduction to Blending Modes ❖ Blending Modes, Opacity & Fill Creating & Modifying Text
13	Exercise on PHOTO RETOUCHING	PHOTO RETOUCHING	❖ Using The Red Eye Tool ❖ The Clone Stamp Tool ❖ The Patch Tool & the Healing Brush Tool ❖ The Spot Healing Brush Tool ❖ The Color Replacement Tool ❖ The Toning & Focus Tools Painting with History
14	Exercise on CREATING SPECIAL EFFECTS	CREATING SPECIAL EFFECTS	❖ Getting Started with Photoshop Filters ❖ Smart Filters ❖ Creating Text Effects Applying Gradients to Text
15	Exercise on Photo Shop Credits	EXPORTING YOUR WORK	❖ Saving with Different File Formats ❖ Saving for Web & Devices ❖ Printing Options ❖ Photo shop Credits
16	Exercise on Logo Creation	Logo Creation	❖ To apply all the tools ❖ Prepare college logo ❖ Prepare logo for industry

Course Code	Course Title	No. of periods/week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM308L	**Data Structures Through Python Lab**	**6**	**90**	**40**	**60**	**2**

S.No.	Chapter/Unit Title	No. Of Periods	CO's Mapped
1.	Introduction to object oriented concepts and implementation of OOP features, Sequential data structure Array-operations on arrays, built in data structures like lists, Tuple, dictionary and sets	21	CO1
2.	Searching and sorting techniques, Linked Storage Representation-Linked Lists	30	CO2
3.	Linear Data Structures-Stacks	18	CO3
4.	Linear Data Structures-Queues	9	CO4
5.	Non Linear Data Structures-Trees,	12	CO5
	Total Periods	90	

	At the end of the course students shall be able to
COURSE OBJECTIVES	i) know the object oriented concepts of python ii) know the various types of Data Structures iii) To familiarize with the representation of Data Structures iv) To use various Data structures in organizing data v) To reinforce theoretical concepts by writing relevant programs

COURSE OUTCOMES:

CO1	CM-308.1	Develop Python programs on object oriented concepts, inbuilt data structures and array operations
CO2	CM-308.2	Develop Python programs on searching, sorting and various Linked Lists and Linked List operations.

CO3	CM-308.3	Develop Python programs on the operations of Stack data structure and its applications
CO4	CM-308.4	Develop Python programs on the operations of Queue data structure
CO5	CM-308.5	Develop Python programs on the operations of Binary Trees

LEARNING OUTCOMES:

1. Exercise on Class and Object in Python
2. Exercise on Overloading concept
3. Exercise on Overriding concept
4. Exercise on Inheritance techniques
5. Exercise on Array operations using Numpy package
6. Exercise on Lists, Tuples, Dictionaries and Sets in Python.
7. Exercise on BUBBLE SORT using Functions.
8. Exercise on SELECTION SORT using Functions.
9. Exercise on INSERTION SORT using Functions.
10. Exercise on QUICK SORT using Functions.
11. Exercise on LINEAR SEARCH using Functions.
12. Exercise on BINARY SEARCH without RECURSION.
13. Exercise on BINARY SEARCH with RECURSION.
14. Exercise on SINGLY LINKED LIST with insert, delete and display operations
15. Exercise on DOUBLY LINKED LIST with insert, delete and display operations
16. Exercise on STACK with push, pop and display operations using built in functions
17. Exercise on STACK with push, pop and display operations without using built in functions.
18. Exercise on Conversion of In-fix expression to post-fix expression using STACKS.
19. Exercise on Evaluation of post-fix expression using STACKS.
20. Exercise on QUEUE with enqueue, dequeue and display operations using arrays.
21. Exercise on QUEUE with enqueue, dequeue and display operations using inbuilt operations.
22. Exercise on BINARY SEARCH TREE with insertion, deletion of nodes.
23. Exercise on BINARY SEARCH TREE with various tree traversal operations.

CO-PO/PSO MATRIX:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM-308.1	3	3	3	2	3	3	3	2	2	2
CM-308.2	3	3	3	1	2	2	2	1	1	1
CM-308.3	3	3	3	2	3	3	2	3	3	3
CM-308.4	3	3	3	2	2	2	2	2	2	2
CM-308.5	3	3	3	2	2	2	2	2	2	2
Average	3	3	3	1.8	2.4	2.4	2.2	2	2	2

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

DATA STRUCTURES THROUGH PYTHON LAB OBJECTIVES AND KEY COMPETENCIES			
Sl.No	Name of the Experiment	Objectives	Key Competencies
1	Exercise on Class and Object in Python	Write a Python program for i. Implementing Class ii. Creating an object iii. Accessing data members and member functions	❖ Correct syntactical errors ❖ Debug logical errors ❖ Observe whether Class and object are properly implemented ❖ Check data members and member functions are properly accessed
2	Exercise on Overloading concepts	Write a Python program for Implementing overloading of functions	❖ Correct syntactical errors ❖ Debug logical errors ❖ Observe whether overloading of functions is done with different signatures
3.	Exercise on Overriding concepts	Write a Python program for Implementing overriding of functions	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether the overriding of functions is done properly
4.	Exercise on Inheritance techniques	Write a Python program for Implementing various inheritance techniques	❖ Correct syntactical errors ❖ Debug logical errors ❖ Observe whether all inheritance types are properly implemented ❖ Check whether the characteristics of each inheritance are achieved or not.
5.	Implement Array	Write a Python program for	❖ Correct syntactical

DATA STRUCTURES THROUGH PYTHON LAB OBJECTIVES AND KEY COMPETENCIES			
Sl.No	Name of the Experiment	Objectives	Key Competencies
	operations using Numpy package	i. Implementing Arrays using Numpy ii. Array addition iii. Array multiplication	errors ❖ Debug logical errors ❖ Check whether Numpy package is properly installed ❖ Observe whether Array addition is properly implemented ❖ Check whether Array Multiplication is performed
6.	Exercise on List, Tuple, Dictionary and set data structures in Python.	Write a Python program for i. Creating a List, Adding elements to list, Accessing elements to list ii. To implement Tuple and get the specified position element from beginning and ending of the Tuple iii. To create a dictionary and perform the search operation for a specific Element. iv. To create a Set and perform set operations like Union, Intersection and Set Difference.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Observe whether List is properly implemented ❖ Check whether the tuple is created and specified position element is accessed ❖ Check whether the Dictionary is created and search operation is performed ❖ Check whether the Set is created and Set operations are properly performed.
7.	Exercises on BUBBLE SORTING using Functions	Write a Python program for i. Implementing Bubble Sort ii. Print the proper result for sorting the elements	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether Bubble sort algorithm is properly implemented ❖ Observe the result for sorting the elements using Bubble sort technique.
8.	Exercise on SELECTION SORT using Functions	Write a Python program for i. Implementing Selection Sort ii. Print the proper result for sorting the elements.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether Selection sort algorithm is properly implemented ❖ Observe the correctness of result for sorting the elements using selection sort

DATA STRUCTURES THROUGH PYTHON LAB OBJECTIVES AND KEY COMPETENCIES			
Sl.No	Name of the Experiment	Objectives	Key Competencies
9.	Exercise on SELECTION SORT using Functions	Write a Python program for I. Implementing Insertion Sort II. Print the proper result for sorting the elements.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether Insertion sort algorithm is properly implemented ❖ Observe the correctness of result for sorting the elements using insertion sort
10.	Exercise on Quick SORT using Functions	Write a Python program for I. Implementing Quick Sort II. Print the proper result for sorting the elements	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether Quick sort algorithm is properly implemented ❖ Observe the correctness of result for sorting the elements using Quick sort
11.	Exercise on Linear search using functions	Write a Python program for I. Implementing Linear Search II. Print the proper result for successful and unsuccessful Linear search	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether Linear Search algorithm is properly implemented ❖ Observe the Base Condition ❖ Observe the result for the search element is present in the list ❖ Observe the result for the search element is not present in the list
12.	Exercise on binary search without Recursion	Write a Python program for I. Implementing Binary Search II. Print the proper result for successful and unsuccessful Binary search	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether Binary Search algorithm is properly implemented ❖ Observe the Base Condition ❖ Observe the result for the search element is present in the list ❖ Observe the result for the search element is not present in the list
13.	Exercise on binary search with	Write a Python program for I. Implementing Binary Search	❖ Correct syntactical errors ❖ Debug logical errors

DATA STRUCTURES THROUGH PYTHON LAB OBJECTIVES AND KEY COMPETENCIES			
Sl.No	Name of the Experiment	Objectives	Key Competencies
	Recursion	II. Print the proper result for successful and unsuccessful Binary search	❖ Check whether Binary Search algorithm is properly implemented ❖ Observe the Base Condition ❖ Observe the result for the search element is present in the list ❖ Observe the result for the search element is not present in the list
14.	Exercises on creation, insertion, deletion, display of elements in a single linked lists	Write a Python program for i. Creation of linked list ii. Inserting an element in Linked list iii. Check for deletion of a node if no element is present and print error message iv. Delete an element from the Linked list v. Display all the elements from the linked list	❖ Rectify syntactical errors ❖ Debug logical errors ❖ Study node structure ❖ Validate whether the memory allocation is done for the node ❖ Confirm whether the addition of node is done at the end ❖ Correct if deletion of an element in an empty list ❖ Confirm whether deletion of required node is done ❖ Observe whether all the elements of the linked list are displayed in proper order
15.	Exercises on creation, insertion, deletion, display of elements in Double linked lists	Write a Python program for I. Creation of doubly linked list II. Inserting an element in Doubly Linked list III. Check for deletion of a node if no element is present and print error message IV. Delete an element from the Linked list V. Display all the elements from the linked list	❖ Rectify syntactical errors ❖ Debug logical errors ❖ Study node structure ❖ Validate whether the memory allocation is done for the node ❖ Confirm whether the addition of node is done at the end ❖ Correct if deletion of an element in an empty list ❖ Confirm whether deletion of required node is done ❖ Observe whether all the elements of the linked list are displayed in proper order
16.	Writea program to Implement a	Write a Python program for	❖ Correct syntactical errors

DATA STRUCTURES THROUGH PYTHON LAB OBJECTIVES AND KEY COMPETENCIES			
Sl.No	Name of the Experiment	Objectives	Key Competencies
	stackusingbuilt-infunctions	i. Creation of Stack consisting of elements using built in functions ii. Insertion of new element is done by () function call iii. Deletion of last element is done by pop() function call iv. Print error message for 'empty stack' if no elements are present for pop() function call v. Print error message for 'stack full' if number of elements exceed size of Stack array	❖ Debug logical errors ❖ Observe declaration of stack using built-in functions ❖ Validate whether a new element is inserted at the top by push() function call ❖ Check whether only the top element is deleted by using negative indexing. ❖ Verify for empty stack condition in pop()
17.	Write à program to implement stack using Arrays	Write a Python program for i. Creation of Stack consisting of elements using arrays ii. Insertion of new element is done by push() function call iii. Deletion of last element is done by pop() function call iv. Print error message for 'empty stack' if no elements are present for pop() function call	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study node structure ❖ Observe declaration of stack using arrays ❖ Validate whether a new element is inserted at the top by push() function call ❖ Check whether only the top element is deleted by pop() function call ❖ Verify for empty stack condition in pop()
18.	Write a program for conversion of infix arithmetic expression into postfix expression	Write a Python program for I. Conversion of infix expression into postfix expression using stacks concept II. Printing the postfix expression	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether the final expression is postfix expression or not.
19.	Write a program for Evaluation of post-fix expression using STACKS.	Write a Python program for i. Evaluation of post-fix expression using STACKS ii. Printing the evaluated result	❖ Correct syntactical errors ❖ Debug logical errors ❖ Check whether the result is correctly evaluated or not.
20.	Write a program to implement a queue using Arrays.	Write a Python program for i. Creation of Queue consisting of elements using arrays ii. Insertion of new element is	❖ Correct syntactical errors ❖ Debug logical errors ❖ Observe declaration of Queue using arrays ❖ Validate whether a new

DATA STRUCTURES THROUGH PYTHON LAB OBJECTIVES AND KEY COMPETENCIES			
Sl.No	Name of the Experiment	Objectives	Key Competencies
		done by add_Queue() iii. Print error message for 'empty queue' if no elements are present for deletion of an empty queue. iv. Print error message for 'queue full' if number of elements exceed size of Queue array upon insertion of new element. v. Deletion of first element is done by delete_Queue()	element is inserted at the end of the array by add_Queue() ❖ Verify for empty Queue condition for deletion of an element ❖ Verify for Queue full condition upon insertion of a new element ❖ Check whether only the first element is deleted by delete_Queue()
21.	Write a program to implement a queue using built in functions	Write a Python program for i. Creation of Queue consisting of elements using inbuilt functions in Python ii. Insertion of new element is done by put() method iii. Print error message for 'empty queue' if no elements are present for deletion of an empty queue. iv. Deletion of first element is done by get() method	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study node structure ❖ Validate whether a new element is inserted at the end of the Queue by put() ❖ Verify for empty Queue condition for deletion of an element using empty() method ❖ Check whether only the first element is deleted by get()
22.	Write a program to implementBINARY SEARCH TREE with insertion, deletion of nodes	Write a Python program for i. Creation of Binary Trees ii. Insertion of a node iii. Deletion of a node	❖ Correct syntactical errors ❖ Debug logical errors ❖ Observe proper definition of elements in a Binary Search Tree ❖ Check whether the node is properly inserted in the Binary Tree ❖ Check whether the node is properly deleted in the Binary Tree ❖ Observe the root node after deleting root node element

DATA STRUCTURES THROUGH PYTHON LAB OBJECTIVES AND KEY COMPETENCIES			
Sl.No	Name of the Experiment	Objectives	Key Competencies
23.	Write a program to implementBINARY SEARCH TREE with various tree traversal techniques.	Write a Python program for I. 1.Creation of Binary Trees II. 2.Perform In-order Traversal of the binary tree III. 3.Perform Pre-order Traversal of the binary tree IV. 4.Perform Post-order Traversal of the binary tree	❖ Correct syntactical errors ❖ Debug logical errors ❖ Validate whether the Tree in-order traversal is properly done ❖ Validate whether the Tree pre-order traversal is properly done ❖ Validate whether the Tree post-order traversal is properly done

UNIT TEST 1	FROM LEARNING OUTCOMES 1 TO 11
UNIT TEST 2	FROM LEARNING OUTCOMES 12 TO 23

Course Code	Course Title	No. of periods/week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM309L	**DBMS Lab**	**6**	**90**	**40**	**60**	**2**

S. No	CHAPTER/UNIT TITLE	NO. OF PERIODS	CO'S
1	Concepts of DBMS & RDBMS	12	CO1
2	Concepts of SQL	24	CO2
3	Basics of PL/ SQL	24	CO3
4	Advance PL/SQL	18	CO4
5	Concepts of NoSQL & Mongo DB.	12	CO5
	TOTAL	90	

COURSE OBJECTIVES	Upon completion of the course the student shall able to learn: Insert, update, delete and select data into/from Relation Database. Develop PL/SQL programs. Insert, update, delete and select data from Mongo DB.

COURSE OUTCOMES:

CO1	CM-309.1	Develop SQL Queries to Create, modify and drop tables and Queries to Insert, update and delete data from tables.
CO2	CM-309.2	Execute SQL Queries to display data on different conditions from different tables
CO3	CM-309.3	Execute PL/SQL Programs
CO4	CM-309.4	Demonstrate the usage of cursors and triggers
CO5	CM-309.5	Execute commands to Insert, update, delete and select data in NOSQL and Mongo DB databases

LEARNING OUTCOMES:

1 Know installation of Oracle

2 Exercise on creating tables.

3 Exercise on inserting records

4 Exercise on updating records

5 Exercise on modifying the structure of the table

6 Exercise on Select command

7 Exercise on querying the table using clauses like WHERE, ORDER BY,

IN, AND, OR,NOT, IS NULL

8 Exercise on GROUP BY, HAVING

9 Exercise on Number functions, character functions, conversion functions and date functions, group functions

10 Exercise on set operators

11 Exercise on sub queries

12 Exercise on Joins

13 Exercise on various date and number format models

14 Exercise on creating tables with integrity constraints

15 Write programs using PL/SQL control statements

16 Exercise on Procedures

17 Exercise on Functions

18 Exercise on pre-defined exception

19 Exercise on user-defined exception

20 Exercise on Cursors

21 Exercise on Triggers

22 Exercise on Installation of Mongo DB

23 Exercise on Creation and Dropping of Database

24 Exercise on Creation and Dropping of Collections.

25 Exercise on Commands of Mongo DB- Insert, update , find, delete and sorting of Documents.

Mini Project: Student has to develop a Mini project applying the skills acquired from the learning outcomes of this course.

CO-PO/PSO MATRIX:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM-309.1	2		3			2	3	3	2	
CM-309.2	2	2	1			2			2	
CM-309.3	2		1					2		2
CM-309.4	2	2	3	3	3	3		2	2	2
CM-309.5	2	3		3		3	3	2	2	
Average	2	2.3	2	3	3	2.6	3	2.3	2	2

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

KEY COMPETENCIES:

S.No	Name of the Experiment	Objectives	Key Competencies
1	Know installation of Oracle	Perform the following: i. To identify the version of Oracle being installed ii. To understand the RAM and HDD requirements for Oracle installation iii. To comprehend the installation steps correctly iv. Setting up of Oracle Administrative Password v. Configuring the Oracle database after post-installation steps of Oracle viz configuring administrative rights for performing vi. To login to Oracle as administrator account and Oracle user account	❖ Observe Oracle version being installed ❖ Observe the RAM & HDD requirements ❖ Rectify for any Oracle installation errors ❖ Able to login as Administrator and as Oracle user account
2	Exercise on creating tables	Perform the following: i. To login with Oracle user account ii. To give correct syntax for table creation iii. To give correct data type for the required fields with appropriate size iv. To display the structure of the table	❖ Correct Table creation syntax errors ❖ Correct the wrong data types and inappropriate sizes for the respective fields ❖ Check for displaying the structure of the table

S.No	Name of the Experiment	Objectives	Key Competencies
3	Exercise on inserting records	Perform the following: i. Check for the required table present already ii. To insert the records correctly iii. To display the records correctly	❖ Correct syntax errors for Insertion of record ❖ Check for insertion of proper values for the required fields ❖ Verify the correct values pertaining to the record are inserted in the required table ❖ Check for displaying of the records correctly
4	Exercise on updating records	Perform the following: i. Check for the required table present already ii. To update the records correctly iii. To display the updated records	❖ Correct syntax errors for update of record ❖ Check for update ❖ Check for displaying of the updated records correctly
5	Exercise on modifying the structure of the table	Perform the following i. To identify the required table present in the system already ii. To add new column iii. To display the records correctly	❖ Correct syntax errors in modifying the structure of the table ❖ Check whether required field is newly added to the existing table ❖ Check for displaying of the modified table correctly
6	Exercise on SELECT command	Perform the following i. To identify the required table present already ii. To display the records in the required table	❖ Check for syntax error in usage of Select command ❖ Check whether Select command is given correctly to display all the records

S.No	Name of the Experiment	Objectives	Key Competencies
7	Exercise on querying the table using clauses like WHERE, ORDER, IN,AND, OR,NOT, IS NULL	Perform the following: i. To use the Select command ii. To use the clauses WHERE, ORDER, IN,AND, OR, NOT, IS NULL along with Select command on the given records in the table	❖ Check for syntax error in usage of Select command with appropriate clauses ❖ Check whether Select command along with appropriate clause is given correctly for the required condition ❖ Check the usage of clauses WHERE, ORDER, IN,AND, OR, NOT along with Select command appropriately
8	Exercise on GROUP BY, HAVING	Perform the following: i. To use the Select command To use the clauses GROUP BY, HAVING along with Select command on the given records in the table	❖ Check for syntax error in the usage GROUP BY, HAVING ❖ Check for usage of GROUP BY, HAVING ❖ Verify output values based on certain condition on few records
9	Exercise on Number functions, character functions, conversion functions and date functions, group functions	Perform the following i. To use functions ii. To use set command along with WHERE condition, GROUP BY, HAVING	❖ Check for syntax error of various functions ❖ Check for usage of various functions ❖ Verify output values based on certain condition on few records
10	Exercise on SET operators	Perform the following iii. To use set command iv. To use set command along with WHERE condition	❖ Check for syntax error in the usage of SET command ❖ Check for usage of SET command for updating values based on certain condition on few records
11	Exercise on sub queries	Perform the following i. To use Select command ii. To use appropriate Operators IN	❖ Check for the syntax error in usage of sub queries ❖ Check for the correctness of the usage of appropriate operators used

S.No	Name of the Experiment	Objectives	Key Competencies
12	Exercise on Joins	Perform the following i. To create two tables ii. To use the common field if two tables are used iii. To know different types of Join	❖ Check for the correctness of the syntax used for joining ❖ Check if the join is created between two tables ❖ Check if self join is created
13	Exercise on various date and number format models	Perform the following: i. To use date formats correctly ii. To use number formats correctly	❖ Check for the syntax of the date formats ❖ Check for the syntax of the number formats
14	Exercise on creating tables with integrity constraints	Perform the following i. Create Primary key ii. Create Foreign key or referential integrity constraint iii. Create NOT NULL constraint iv. Create UNIQUE Key constraint v. Create CHECK constraint	❖ Check for the syntax errors in usage of all types of Integrity constraints ❖ Check whether different types of Integrity constraints are used
15	Write programs using PL/SQL control statements	Perform the following i. To use IF .. ELSE statements ii. To use iterative statements – Simple loop, While Loop, For Loop	❖ Check for the syntax of IF.. ELSE statements ❖ Check for the syntax of all iterative statements
16	Exercise on Procedures	Perform the following i. To know the concept of stored procedures ii. To declare procedures iii. The type of parameters IN,INOUT,OUT iv. To call procedures from other procedures	❖ Check for proper declaration of procedures ❖ Check for syntax ❖ Check for proper calling of procedures

S.No	Name of the Experiment	Objectives	Key Competencies
17	Exercise on Functions	Perform the following i. To know the concept of stored functions ii. To declare function with return data iii. To call functions from other functions	❖ Check for proper declaration of function ❖ Check for syntax of parameters and its data type ❖ Check for proper return data type from the functions ❖ Check for variable assignment to get the returned value from the function
18	Exercise on pre defined exception	Perform the following i. To know the concept of pre defined exception ii. To write programs with exception section	❖ Check for proper declaration Exception section ❖ Check for syntax of Exception section
19	Exercise on user defined exception	Perform the following i. To know the concept of user defined exception ii. To write programs with exception section	❖ Check for proper declaration Exception section ❖ Check for syntax of Exception section
20	Exercise on Cursors	Perform the following i. To know the concept cursors ii. To know the fetch data from database	❖ Check for the syntax of cursor ❖ Check for open cursor, fetch data, close cursor ❖ Check for the result
21	Exercise on Triggers	Perform the following i. To know the concept of triggers ii. Validation before and after insert, before and after update, before and after delete data	❖ Check for the syntax of trigger ❖ Write a trigger which raises before insert data ❖ Raise trigger ❖ Repeat the procedure for remaining ❖ Check for the result

S.No	Name of the Experiment	Objectives	Key Competencies
22	Exercise on Installation of Mongo DB	Perform the following i. To download and install Mongo DB	❖ Observe Mongo DB version being installed ❖ Observe the RAM & HDD requirements ❖ Rectify for any MongoDB installation errors Able to login as Administrator
23	Exercise on Creation and Dropping of Database	Perform the following i. Create the Database ii. Drop the Database	❖ Know the use of create Database() and dropDatabase() ❖ Correct Database creation syntax errors ❖ Check for displaying the database name
24	Exercise on Creation and Dropping of Collections	Perform the following i. Create the Collection ii. Drop the Collection	❖ Know the use of create Collection() and drop() ❖ Correct Database creation syntax errors ❖ Check for collection name ❖ Check for the collection dropped
25	Exercises on commands of Mongo DB	Execute the following commands of Mongo DB i. Insert the Document ii. update the Document iii. find the Document iv. Delete the Document v. sort the Documents	❖ Know the syntax of insert(), update(), find(), remove(), sort() functions. ❖ Correct syntax errors. ❖ Check out for different input values.

UNIT TEST 1	FROM LEARNING OUTCOMES 1 TO 12
UNIT TEST 2	FROM LEARNING OUTCOMES 13 TO 25

Course code	Course Title	No. of Periods/Weeks	Total No. of periods	Marks for FA	Marks for SA
26CM310L	Linux Practicals	4	60	40	60

TIME SCHEDULE

S.No.	Chapter/Unit Title	No. of Periods	CO's Mapped
1	Linux Installation and user management	8	CO1
2	LINUX commands	24	CO2
3	LINUX editors	4	CO3
4	LINUX Shell Programs	24	CO4
TOTAL		60	

COURSE OBJECTIVES

Upon completion of the course the student shall be able to	
(i)	To know the basics of linux installation and user management.
(ii)	To use and execute linux commands.
(iii)	To know the usage of different Linux editors.
(iv)	To know about the shell programming and execute shell programs.

COURSE OUTCOMES

CO1	CM310.1	Demonstrate Linux installation and user management
CO2	CM310.2	Demonstrate Linux commands and Linux Editors
CO3	CM310.3	Execute shell programs using expressions, operators
CO4	CM310.4	Execute shell programs on control statements and conditional statements, Files, Functions and string manipulations.

LEARNING OUTCOMES

S.No	LEARNING OUTCOMES	THEORY (explained using PPT,useful videos from youtube etc.,)	PRACTICAL (Key competencies table enclosed)
1	Installation of Linux	Explanation of Linux installation like ubuntu,kali,centos	Demonstration of Installation of various Linux distributions like Ubuntu, kali,Centos etc.,
2	Creating and managing user accounts in LINUX	Explanation of Creating and managing user accounts in LINUX	Demonstration of Creating and managing user accounts in LINUX
3	Exercise on Basic Commands	Listing and explaining basic Linux Commands	Demonstration on Basic Linux commands
4	Exercise on Linux File Directory Management commands	Listing and explaining of Linux File Directory Management commands	Demonstration of Linux File Directory Management commands
5	Exercise on Linux Permissions and Ownership commands	Listing and explaining of Linux Permissions and Ownership commands	Demonstration of Linux Permissions and Ownership commands
6	Exercise on Linux Archiving and Compression commands	Listing and explaining of Linux Archiving and Compression commands	Demonstration of Linux Archiving and Compression commands
7	Exercise on Linux System Monitoring and Management commands	Listing and explaining of Linux System Monitoring and Management commands	Demonstration of Linux System Monitoring and Management commands
8	Exercise on Linux Miscellaneous and Utility Commands	Listing and explaining of Linux Miscellaneous and Utility commands	Demonstration of Linux Miscellaneous and Utility commands
9	Exercise on Linux Disk and Memory Management commands	Listing and explaining of Linux Disk and Memory Management commands	Demonstration of Linux Disk and Memory Management commands
10	Exercise on Linux filters commands	Listing and explaining of on Linux filters commands commands	Demonstration of on Linux filters commands commands
11	Exercise on Linux Text editors	Explanation of Linux editors like vi,vim etc.,	Demonstration of Linux editors like vi,vim
12	Exercise on shell script using expressions,	NA	Execution of shell script

	operators		
13	Exercise on Shell Script on control statements.	NA	Execution of shell script
14	Exercise on Shell script on conditional statements.	NA	Execution of shell script
15	Exercise on Shell script on LOOP statements	NA	Execution of shell script
16	Exercise on shell script on string manipulations.	NA	Execution of shell script
17	Exercise on Shell script on FUNCTIONS	NA	Execution of shell script
18	Exercise on Shell script on FILES	NA	Execution of shell script

CO-PO-PSO Mapping Matrix

CO Number	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CM310.1	2	3	1	1	2	1	3	2	1
CM310.2	1	3	2	1	3	2	2	3	2
CM310.3	1	2	2	2	2	1	2	3	2
CM310.4	1	3	3	2	3	2	3	3	3
Average	1.25	2.75	2	1.5	2.5	1.5	2.5	2.75	2

KEY COMPETENCIES

Sl.No	Name of the Experiment	Objectives	Key Competencies
1	Installation of Linux	Installation on Linux distributions like Ubuntu, kali,Centos etc.,	1. Choose method to install 2. Choose a Linux distribution 3. Check for the syntax errors and clear the errors 4. Follow install method instructions
2	Creating and managing user accounts in LINUX	Creating and managing user accounts in Linux	1. Create user account 2. Modify user accounts 3. Delete user accounts 4. Remove user accounts
3	Exercise on Basic Commands	To Run Linux shell basic commands	1. Open Linux OS 2. Open shell 3. Run basic shell commands 4. Observe output of command
4	Exercise on Linux File Directory Management commands	To Run Linux File Directory Management commands	1. Open Linux OS 2. Open shell 3. Run various File directory and management commands in the shell 4. Observe output of command
5	Exercise on Linux Permissions and Ownership commands	To Run Linux Permissions and Ownership commands	1. Open Linux OS 2. Open shell 3. Run various Permissions and Ownership command in the shell 4. Observe output of command
6	Exercise on Linux Archiving and Compression commands	To Run Linux Archiving and Compression commands	1. Open Linux OS 2. Open shell 3.Run various Archiving and Compression command in the shell 4.Observe output of command
7	Exercise on Linux System Monitoring and Management commands .	To Run System Monitoring and Management commands	1. Open Linux OS 2. Open shell 3. Run various System Monitoring and Management commands in the shell 4. Observe output of command

8	Exercise on Linux Miscellaneous and Utility Commands	To Run Miscellaneous and Utility Commands	1. Open Linux OS 2. Open shell 3. Run various Miscellaneous and Utility Commands in the shell 4. Observe output of command
9	Exercise on Linux Disk and Memory Management commands	To Run Disk and Memory Management commands	1. Open Linux OS 2. Open shell 3. Run various Disk and Memory Management Commands in the shell 4. Observe output of command
10	Exercise on Linux filters commands	To Run Linux filters commands	1. Open Linux OS 2. Open shell 3. Run various Filtering Commands in the shell 4. Observe output of command
11	Exercise on Linux Text editors	Working on Linux text editors like vi	1.Opening/Creating a File 2. Editing Modes (for modal editors like Vim) 3. Text Input and Manipulation 4. Saving Changes 5. Exiting editor
12	Exercise on shell script using expressions, operators	Write a shell script program on expressions and operators to demonstrate various operators and expression in Linux shell script	1. Create the Shell Script File with operators and expressions 2. Add the Shebang Line 3. Implement the expressions and operators 4. Check for the correctness of output for the given input 5.Save script 6. Make the Script Executable by granting execute permissions 7.Run the Script 8. Verify and Debug
13	Exercise on Shell Script on control statements.	Write a shell script program on control statements to demonstrate various control statements in Linux shell script	1. Create the Shell Script File with control statements 2. Add the Shebang Line 3. Implement the expressions and operators 4. Check for the correctness of output for the given input 5.Save script 6. Make the Script Executable by granting execute permissions 7.Run the Script 8. Verify and Debug

14	Exercise on Shell script on conditional statements.	Write a shell script program on conditional statements to demonstrate various conditional statements in Linux shell script	1. Create the Shell Script File with conditional statements 2. Add the Shebang Line 3. Implement the expressions and operators 4. Check for the correctness of output for the given input 5.Save script 6. Make the Script Executable by granting execute permissions 7.Run the Script 8. Verify and Debug file
15	Exercise on Shell script on LOOP statements	Write a shell script program on Loop statements to demonstrate various loop statements in Linux shell script	1. Create the Shell Script File with loop statements 2. Add the Shebang Line 3. Implement the expressions and operators 4. Check for the correctness of output for the given input 5.Save script 6. Make the Script Executable by granting execute permissions 7.Run the Script 8. Verify and Debug file
16	Exercise on shell script on string manipulations.	Write a shell script program on string manipulation commands to demonstrate string manipulation commands in Linux shell script	1. Create the Shell Script File with string commands 2. Add the Shebang Line 3. Implement the expressions and operators 4. Check for the correctness of output for the given input 5.Save script 6. Make the Script Executable by granting execute permissions 7.Run the Script 8. Verify and Debug file
17	Exercise on Shell script on FUNCTIONS	Write a shell script program on functions to demonstrate functions in Linux shell script	1. Create the Shell Script File with functions 2. Add the Shebang Line 3. Implement the expressions and operators 4. Check for the correctness of output for the given input 5.Save script 6. Make the Script Executable by granting execute permissions 7.Run the Script 8. Verify and Debug file

18	Exercise on Shell script on FILES	Write a shell script program on files to demonstrate files in Linux shell script	1. Create the Shell Script File 2. Add the Shebang Line 3. Implement the expressions and operators 4. Check for the correctness of output for the given input 5.Save script 6. Make the Script Executable by granting execute permissions 7.Run the Script 8. Verify and Debug file

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TEST-I & UNIT TEST-II

Unit Test	Key Competencies to be Covered
Unit Test – 1	From 1 to 9
Unit Test – 2	From 10 to 18

FOURTH SEMESTER								
THEORY SUBJECTS	COMPULSORY SUBJECTS	**S No**	**Sub. Code**	**Subject**	**PRACTICUM**	**Periods**	**Total Periods**	**Credits**
		1	26CM401T	WEB TECHNOLOGIES	N	6	90	3.5
		2	26CM402T	OOP THROUGH JAVA	N	6	90	3.5
		3	26CM403T	COMPUTER NETWORKS	N	6	90	3.5
	ELECTIVE SUBJECTS	ELECTIVE SUBJECT						
		4	26CM404E	CYBER SECURITY	N	3	45	2
		5	26CM405E	CLOUD COMPUTING	N			
	AUDIT SUBJECT							
		6	26CM406A	TROUBLESHOOTING OF COMPUTER NETWORKS	Y	2	30	
		TOTAL THEORY + PRACTICUM PERIODS				**23**	**345**	**12.5**
PRACTICAL SUBJECTS	COMPULSORY SUBJECTS	7	26CM407L	WEB TECHNOLOGIES LAB	Y	4	60	2
		8	26CM408L	COMMUNICATION & EMPLOYABILITY SKILLS	N	6	90	2.5
		9	26CM409L	JAVA LAB	N	6	90	2.5
				ACTIVITY	…	3	45	0.5
		TOTAL PRACTICAL + PRACTICUM PERIODS				**19**	**285**	**7.5**
GRAND TOTAL PERIODS (15 WEEKS PER SEMESTER)						**42**	**630**	**20**

WEB TECHNOLOGIES

Course code	Course Title	No. of Periods/Week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM401T	**WEB TECHNOLOGIES**	**6**	**90**	**30**	**70**	**3.5**

TIME SCHEDULE:

S. No.	Chapter/ Unit Title	No. of Periods	Weightage of marks	No. of Short Answer Questions	No. of Essay Questions	CO's Mapped
1.	Introduction to Web Design and HTML	18	22	2	2	CO1
2.	CSS (Cascading Style Sheets) and XML	18	18	2	1.5	CO1, CO2
3.	JavaScript	16	14	2	1	CO2
4.	Angular JS & JQuery	18	18	2	1.5	CO3
5.	Web Servers and PHP — Server-side Scripting	20	22	2	2	CO4
Total		**90**	**94**	**10**	**8**	

COURSE OBJECTIVES

i) Understand the basic elements of web page

ii) Know the working with HTML, CSS

iii) To familiarize the various Technologies like Java Script, J Query, PHP.

iv) To understand Database connectivity Using PHP

COURSE OUTCOMES:

CO1	CM401.1	Implement interactive web page(s) using HTML and CSS
CO2	CM401.2	Know how to format and validate Web page elements using JavaScript and describe data in a web page using XML.
CO3	CM401.3	Understand and apply jQuery to enhance interactivity, handle events, apply visual effects, and extend functionality using plugins
CO4	CM401.4	Build Dynamic web site using server side PHP Programming and database connectivity using PHP.

LEARNING OUTCOMES:

1. Introduction to Web Design and HTML

1.1 Understand Principles of Effective Web Design

1.1.1 Basic web Terminology.

1.1.2 Describe Anatomy of web page.

1.1.3 Understand different Web page elements.

1.1.4 Navigate through web pages

1.1.5 Narrate steps in building web site

1.1.6 Narrate steps in launching

1.1.7 Narrate maintaining web site.

1.2 HTML Introduction

1.2.1 Introduction and Overview of HTML

1.2.2 Discuss the rules for designing a HTML document.

1.2.3 Explain the structure of HTML document.

1.2.4 Define HTML element and Attribute.

1.2.5 Study the basic tags in HTML <html>, <head>, <title>, <body>.

1.2.6 Study the header tags <h1> to <h6>

1.2.7 Discuss the Physical formatting tags <b>, <i>, <u>, <strike>, <sub>, <sup>, big>,

<small>

1.2.8 Discuss the Logical formatting tags <q>, , <cite>, <<ins>, <del>,

1.2.9 Discuss the <marquee> with attributes.

1.2.10 List Character entities.

1.2.11 Explain the List tags like <ul>, <ol>, <li>, <dl>, <menu> with attributes.

1.3 Describe the setting of tables.

1.3.1 Describe the tags <table>, <tr>, <td>, <th>,<tbody>, <thead>, <tfoot>, <colspan>, <rowspan>

1.4 Connecting to hyperlinks and Imaging

1.4.1 Explain the link and imaging tags <a>, <img> with attributes.

1.5 Forms: Input Types, Elements, and Attributes

1.5.1 Build web forms using various <form>, <input>, <button>, <label>, <select>, <options>,

<textarea>, <legend> <fieldsets>with attributes.

1.5.2 Utilize form attributes (name, id, value, required, etc.) for data collection and usability.

1.6 Basic Form Validation Techniques

1.6.1 Implement basic HTML form validation using built-in attributes (required, pattern, minlength)

1.6.2 Identify common input errors and ensure user-friendly feedback is provided.

1.6.3 Explain the importance of validation for both user experience and data integrity.

1.7 Understand the difference between and tags

1.7.1 Recognize that is a block-level element used for grouping large sections.

1.7.2 Organize web page components (header, footer, sidebar, main content) using tags.

1.7.3 Use for inline formatting:

1.7.4 Highlight or style specific words or phrases inside larger text blocks.

2. CSS (Cascading Style Sheets) and XML

2.1 Illustrate about cascading style sheets

2.1.1 Understand the level of styles inline, internal and external style sheets.

2.1.2 Explain ID and Class selectors in CSS

2.1.3 Explain about Color and background properties

2.1.4 Explain about Box properties like Border, position, margin, padding of elements.

2.1.5 Understand the use of different CSS display properties to control how elements appear on a webpage.

2.1.6 Apply CSS positioning methods to place elements precisely on a webpage.

2.1.7 Use the float property to align elements (like images or boxes) to the left or right within a container

2.1.8 Create flexible and responsive layouts using the Flexbox model to align and distribute space among items in a container.

2.1.9 Design complex webpage layouts using the CSS Grid system to create rows and columns easily.

2.1.10 Apply CSS styles to improve the appearance and usability of HTML forms and

tables.

2.1.11 Enhance user experience by adding smooth visual effects using CSS transitions and basic animations.

2.2 Understand XML

2.2.1 Explain the Purpose of XML

2.2.2 Describe how to organize data in the form of XML

2.2.3 Explain the rules for designing XML document.

2.2.4 Understand the significance of Namespace.

2.2.5 List the various applications of XML.

3. JavaScript

3.1 Types of scripting-JavaScript

3.2 Differentiate between Client-side and Server-side scripting.

3.3 List Client side and server side scripting languages.

3.4 Describe the features of Java Script.

3.5 Placing JavaScript code in HTML.

3.6 Understand functions

3.6.1 Know how to define and call a function.

3.6.2 Know how to pass parameters.

3.6.3 Understand the purpose of getElementById and getElementByName method

3.6.4 Describe the global functions provided by JavaScript.

3.7 Form Handling in Java Script

3.8 Illustrate Arrays

3.8.1 Understand single and multi-dimensional arrays.

3.8.2 Design small programs using arrays.

3.9 Various Objects provided by JavaScript.

3.9.1 Math object.

3.9.2 String object.

3.9.3 Date object.

3.9.4 Boolean and Number object.

3.10 Describe events in java script.

4 Angular JS and JQuery

4.1 Angular JS

4.1.1 What is AngularJS? Architecture, Advantages & Features.

4.1.2 List and Explain Angular JS Directives like ng-app,ng-init ,ng-model , ng-repeat

4.1.3 Explain AngularJS Expressions like number, string, object, and array.

4.1.4 Explain AngularJS Filters like lowercase, uppercase, filter, orderby, currency.

4.2 Define JQuery

4.3 List the features of JQuery

4.4 List JQuery plug ins

4.5 Explain the steps to include JQuery in Web Pages

4.6 Explain JQuery Syntax with example program

4.7 Describe the JQuery Selectors

4.8 Accessing HTML elements by using Element Selectors, ID, Class Selectors

4.9 Explain the JQuery Document Ready Event

4.10 Describe the JQuery Event handling methods

4.10.1 Mouse Events

4.10.2 Keyboard Events

4.10.3 Form Events

4.10.4 Document/Window events

4.11 Explain effects of JQuery like hide, show, fade In, fadeout, fade Toggle, fade To, SlideDown, SlideUp, Slide Toggle

4.12 Perform DOM manipulation using jQuery methods .html(), .text(),. val(), .append()

4.13 Implement event handling click, hover, and submit to create interactive web pages.

4.14 Integrate and utilize jQuery plugins to extend web functionality.

5 Web servers and PHP - Server-side scripting.

5.1 Web servers:

5.1.1 Understand the architecture of a Web server.

5.1.2 List various web servers.

5.1.3 Illustrate the various HTTP request types and their difference.

5.1.4 Compare the properties of IIS and Apache.

5.2 Fundamentals of PHP

5.2.1 State the importance of PHP

5.2.2 Explain how to combine HTML and PHP.

5.2.3 Explain how to access HTML, PHP documents from web servers.

5.3 Data types, Variables and Constants

5.3.1 List Data types

5.3.2 Explain Data types with examples

5.3.3 Explain how to declare Variables and Constants.

5.4 List and explain string manipulation functions.

5.5 Understand Arrays

5.5.1 Explain types of arrays.

5.5.2 Design small programs using arrays.

5.6 Explain form handling in PHP

5.6.1 Access elements of form using $_GET,$_POST

5.7 Know how to access My SQL Database

5.7.1 List and explain My SQL database functions in PHP.

5.7.2 Explain the steps of connecting to a Database.

5.7.3 Know about retrieving data from a table.

5.7.4 Know about inserting data into a table.

5.7.5 Know about updating the data in a table.

5.7.6 Know about deleting data from a table.

5.7.7 Design some simple programs to insert, delete, update and retrieve data from database

5.8 Cookies

5.8.1 Define Cookie.

5.8.2 Know how to create and delete a cookie.

5.8.3 Know the purpose of cookie.

5.9 Sessions

5.9.1 Define Session

5.9.2 Understand how to create a session.

5.9.3 Know how to destroy a session.

5.9.4 Know the purpose of session.

5.9.5 Differentiate Sessions and Cookies.

5.10 Passing data from one web page to other webpage using query string.

CO-PO MAPPING

Course Code **CM-401**	**Course Title: Web Technologies** **Number of course outcomes:04**				**No. of periods:90**
POs	**Mapped with CO No.**	**CO Periods Addressing PO in column1**		**Level (1,2,3)**	**Remarks**
		No	%		
PO1	CO1	25	30	2	>40% Level3
PO2	CO2,CO3,CO4	60	70	3	Highly addressed
PO3	CO1,CO2,CO3,CO4	60	70	3	
PO4					25% to 40% Level 2
PO5	CO1,CO2,CO3,CO4	50	60	3	Moderately Addressed
PO6					
PO7	CO2,CO3,CO4	60	70	3	5% to 25% Level1 Low addressed <5% Not addressed

COURSE CONTENT:

1. **Introduction to Web Design and HTML (Hyper Text Markup Language)**

Principles of effective web design, Anatomy of a web page: structure, layout, and basic navigation,

HTML

Introduction to HTML, Format of Web page, Tags and Attributes, Formatting text, Adding images, Positioning. Lists, Colors, Tables. Connecting to hyperlinks and Imaging, Forms: input types, elements, and attributes, Basic form validation techniques, Basic Page Layout using and

2. **CSS (Cascading Style Sheets) and XML**

Introduction, inline, internal, and external styles, CSS selectors, Text and font styling, colors, backgrounds, Box model, Layout techniques: display, position, float, flexbox, grid, Responsive design, Styling forms and tables, Transitions and simple animations.

XML: Purpose of XML, Structure and syntax of XML documents, Data organization, XML

Namespaces, Simple XML use cases.

3. **JavaScript**

Introduction to Scripting, JavaScript features, Functions – Function definitions, Use of getElementById, getElementByName, Global functions, Form handling. Arrays – Declaring and allocating arrays, passing arrays to functions, sorting and Searching arrays, Objects – Math object, String object, Date object, Boolean and Number object, Events in java script.

4. **Angular JS & JQuery**

Angular JS

Introduction to Angular JS, Features, Advantages, Angular JS architecture, Directives, Expressions, Filters, Sample programs.

JQuery

Role of jQuery, Features of J Query, steps to include jQuery in Web Page, Syntax, including document ready function, Selectors: element, class, ID, DOM manipulation with jQuery, jQuery event handling, jQuery effects, working with plugins.

5. **Web Servers and PHP —Server-side Scripting**

Web server: Overview of HTTP protocol and request types, Client-side vs. server-side scripting Types of web servers, hosting websites and accessed, Requesting and serving HTML, PHP documents.

PHP

Introduction to PHP and embedding in HTML, Syntax, variables, datatypes, String Functions, arrays, form Handling with PHP, Sessions and cookies, Connecting to a database, Simple CRUD operations, Passing data between web pages

REFERENCE BOOKS:

1) Principles of Web Design, Sklar, TMH

2) HTML complete reference, Powell, TMH

3) Basics of Web Site Design, NIIT – PHI

4) WWW Design with HTML, Xavier (TMH)

5) Internet & World Wide Web, Dietel and Dietel, Pearson education Asia.

6) Complete Reference PHP, Steven Holzer-McGraw Hill

7) JQuery Cook book, O' Reilly Media

8) www.w3schools.com

9) www.php.net

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TEST-I & UNIT TEST-II

Unit Test	Learning outcomes to be covered
Unit test-1	FROM 1.1 to 3.5
Unit test-2	FROM 3.6 to 5.10

Course code	Course title	No. Of periods/week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM402T	**OOP THROUGH JAVA**	**6**	**90**	**30**	**70**	**3.5**

TIME SCHEDULE

S. No.	Chapter/ Unit Title	No. of Periods	Weightage of marks	No. of Short Answer Questions	No. of Essay Questions	CO's Mapped
1.	Object Oriented Programming concepts and Basics of java, Overloading	17	14	2	1	CO1, CO2
2.	Concepts of Inheritance, Overriding, Interfaces, Packages and I/O streams.	17	22	2	2	CO2
3.	Collection Classes and Exception Handling	16	18	2	1.5	CO3
4.	Multi-threaded Programming. Fundamentals of AWT	18	18	2	1.5	CO4
5.	Event Handling and Swings	22	22	2	2	CO5
	Total	**90**	**94**	**10**	**8**	

COURSE OBJECTIVES

Upon completion of the course the student shall be able to:	
(i)	To know applying object-oriented programming paradigm in problem solving.
(ii)	Able to design multi-tasking application with the knowledge of multi-threading.
(iii)	Able to design data structures concepts in java with collection classes.
(iv)	Familiarized to develop graphical user interface with event handling mechanism using Swings.

COURSE OUTCOMES:

At the end of the course the student able to learn following:		
CO1	CM402.1	Know the object-oriented programming concepts in problem solving. Use syntax and semantics of object-oriented paradigm.
CO2	CM402.2	Design optimized definition for an application with re-usability features and packages in project development.
CO3	CM402.3	Knows the usage of utilities in real time data structures.
CO4	CM402.4	Demonstrate multi-threading concepts to implement multitasking and multi programming applications.
CO5	CM402.5	Demonstrate to design effective dynamic user interface for any front-end applications using Swings and events.

LEARNING OUTCOMES:

1.0 Object oriented programming concepts and Basics of java and over loading

1.1 Know about object-oriented programming

1.2 Compare procedure-oriented programming and object-oriented programming

1.3 List and explain features of object-oriented programming

1.4 Importance of Java in Internet programming.

1.5 Explain features of Java. Define Byte codes of Java, JVM.

1.6 Explain data types, scope and life time of variables, comment statements in Java.

1.7 Describe conversion and casting features.

1.8 Apply one-dimensional and two–dimensional arrays give example programs.

1.9 Illustrates usage of conditional and iteration statements of Java with an example program.

1.10 Describe usage of jump statements, break, and continue statements.

1.11 Describe how to create classes and objects.

1.12 Demonstrate Usage of new operator and methods.

1.13 Explain usage of constructors with an example programs.

1.14 Apply method overloading and construction overloading in applications.

1.15 Describe usage of 'this' pointer with example.

1.16 Usage of command-line arguments.

2.0 Concepts of inheritance, overriding, Interfaces, Packages and I/O Streams

2.1 Explain about string class with methods.

2.2 Explain implementation of inheritance with an example program.

2.3 Illustrate how to implement multilevel inheritance with an example program.

2.4 Explain method overriding and usage of super keyword.

2.5 Describe concept of Interfaces.

2.6 Define an Interface. Differences between abstract classes and interface.

2.7 Explain how to implement interfaces with sample program.

2.8 Define a package. Explain the concept of class path.

2.9 Describe concept of Access protection.

2.10 Illustrate the mechanism of importing packages.

2.11 Give simple application to design packages with sample programs.

2.12 List different types of I/O streams.

2.13 Explain how to read and write data through console input and output streams.

2.14 Explain DataInputStream and DataOutputStreams with example.

2.15 Explain various file access operation by using File-streams.

2.16 Explain sample programs on above streams.

3.0 **Collection classes and Exception Handling.**

3.1 What is collection frame work and Hierarchy of collection frame work.

3.2 List Collection Interfaces and explain the following with examples

3.2.1 List

3.2.2 Set

3.2.3Queue

3.3List Collection classes and explain the following with examples

3.3.1ArrayList

3.3.2LinkedList

3.3.3HashSet

3.4How to access a Collection via an Iterator?

3.5Describe sources of errors. Types of exceptions Checked & Unchecked.

3.6 Apply following key words to handling exceptions through sample programs

3.6.1 Try

3.6.2 Catch

3.6.3 Finally

3.7 Explain concept of multi-catch statements with example.

3.8Explain how to write nested try in exception handling with example.

4.0Multi-threaded programming and Fundamentals of AWT.

4.1 Describe multi-threading.

4.2 Explain Thread life cycle and states

4.3Explain how to Creating single thread with example program
4.4Explain how to Creating multi thread with example program.
4.5Illustrate thread priorities in multiple threads with an example.
4.6Describe the concept of synchronization with example program.
4.7Describe AWT classes. Component class methods
4.8Working with Color Font classes.

5.0 Event Handling and Swings.

5.1Describe Types of Events and Demonstrate event handling mechanism
5.2List and explain different event classes.
5.3List and explain event listener interfaces
5.4Demonstrate handling mouse events with sample program.
5.5Demonstrate handling keyboard events with sample program.
5.6Swing Hierarchy
5.7MVC (Model-View-Controller) pattern in Swing
5.8Explain following Containers in swings with example programs

JFrame, JPanel, JDialog

5.9Explain following Controls with example programs:

JButton JLabel, JTextField JPasswordField, JCheckBox JRadioButton, JComboBox, JList

5.10 Design simple application programs using multiple controls.

CO-PO/PSO MATRIX:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM402.1	**2**	**1**	**1**	**2**	**2**	**2**	**1**	**2**	**2**	**2**
CM402.2	**1**	**3**	**3**	**3**	**1**	**3**	**1**	**1**	**3**	**1**
CM402.3	**3**	**2**	**3**	**2**	**2**	**3**	**1**	**1**	**3**	**3**
CM402.4	**1**	**1**	**3**	**2**	**2**	**3**	**2**	**2**	**3**	**2**
CM402.5	**3**	**3**	**3**	**3**	**2**	**3**	**2**	**2**	**3**	**3**
Average	**2**	**2**	**2.5**	**2.4**	**1.6**	**3**	**1.5**	**1.6**	**3**	**2.2**

3=Strongly mapped, 2=moderately mapped, 1=slightly mapped

- Method is to relate the level of PO with the number of hours devoted to the COs which address the given PO.
- If > 40% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 3.

- If 25 to 40% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 2.
- If 5 to 25% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 1.
- If < 5% of classroom sessions addressing a particular PO, it is considered that PO is not addressed

COURSE CONTENT:

1. Basics of java and overloading: object-oriented programming -Importance of Java to Internet – Byte codes. Features of Java: OOPS concepts –Data types –type conversions – casting – Arrays. Usage of classes – objects – new – methods – constructors – method overloading, string classes – command line arguments-static members-this pointer

2. Concepts Inheritance Overriding Interfaces and Package: -Usages of Inheritance: inheritance super class, sub classes – Multi level inheritance – super keyword - overriding –Abstract Classes-Interfaces-Packages. - I/O Streams-Accessing data through console input and output

3. **Collections, Exception Handling**: -Collection Frame work- Collection Interfaces – Collection Classes-Iterator - Exception handling: Source of errors – error handling – Exception handling-Multi catch statements.

4. Multi-threading Fundamentals of AWT: –- Define thread – life cycle of thread - Multi threading –Synchronization- – Thread properties. -AWT Classes-Color Font classes-Working with Graphics-color-fonts-

5.Swings and Event Handling: Event Classes-Listener interfaces- mouse events -Key Event-Swing Hierarchy-MVC (Model-View-Controller) pattern in Swing-Swing Containers- JFrame, ,JPanel, JDialog-Swaing Controls-Like JButton - JTextField – JradioButton- JComboBox

REFERENCE BOOKS:

1. The complete reference Java -- Pattrick Naughten, Herbert Schildt TMH Company Limited, New Delhi.
2. Programming in JAVA -- P. Radhakrishna, University Press
3. Programming in Java -- Muthu - Thomson
4. Java Foundations of Programming – NIIT, PHI
5. Programming with Java -- Balagurusamy, TMH

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TEST-I & UNIT TEST-II

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 3.4
Unit test-2	From 3.5 to 5.10

Course code	Course Title	No. of Periods / Week	Total No. of periods	Marks for FA	Marks for SA	CREDITS
26CM403T	**Computer Networks**	6	90	30	70	**3.5**

TIME SCHEDULE:

Unit No.	Chapter/Unit Title	No. of Periods	Marks	No. of Short Answer Questions	No. of Essay Type Questions	CO's Mapped
1.	Fundamentals of Networking and Data communication	15	14	2	1	CO1
2.	Physical and Data Link Layer	18	22	2	2	CO2
3.	Network and Transport Layer Concepts	22	22	2	2	CO3
4.	Session, Presentation & Application Layers Concepts	20	22	2	2	CO4
5.	Network Security and Emerging Concepts	15	14	2	1	CO5
Total		**90**	**94**	**10**	**8**	

COURSE OBJECTIVES:

Upon successful completion of this course, the student will be able to:

2. Explain fundamental networking concepts, topologies and functions of OSI and TCP/IP layers.
3. Explain the role of physical and data link layers.
4. Gain knowledge of Network IP addresses and transport layer protocols in the network
5. Describe key protocols and services used in networking.
6. Appreciate the importance of network security, and emerging trends in communication systems.

COURSE OUTCOMES:

CO1	CM403.1	Explain the basic concepts of data communication and networking models
CO2	CM403.2	Describe the roles and functionalities of the Physical and Data Link layers in network communication.
CO3	CM403.3	Apply IP addressing, sub-netting, routing, and transport layer mechanisms

CO4	CM403.4	Illustrate the functions of session, presentation, and protocols of application layers
CO5	CM403.5	Describe network security principles and emerging network technologies.

LEARNING OUTCOMES:

Unit 1: Fundamentals of Networking and Data communication

1.8 Basics of Data Communication: Components, Data Representation, Data Flow: Simplex, Half-Duplex and Duplex.
1.9 Analog and Digital Signals, Periodic and Non-Periodic Signals.
1.10 Define Sine Wave characteristics: Amplitude, Frequency, Phase, Wavelength.
1.11 Digital Signal concepts: Bit-rate and Bit-length.
1.12 Define Transmission Impairment – Attenuation, Distortion and Noise
1.13 Define Performance - Bandwidth, Throughput, Latency, Jitter (Basic concepts only).
1.14 Identify various Transmission Modes – Parallel and Serial Transmission, Asynchronous and Synchronous Transmission
1.15 Overview and classification of networks: LAN, MAN, WAN, PAN
1.16 State the need and benefits of Networking.
1.17 Importance of Network standards and communication models.
1.18 Explain OSI Model: architecture and functions
1.19 Explain TCP/IP Model: architecture and functions
1.20 Comparison of OSI and TCP/IP models
1.21 Explain various network Topologies: Bus, Ring, Star, Mesh, Hybrid – with advantages/disadvantages
1.22 Network Hardware: Repeater, Hub, Switch, Router, NIC, Modem, Gateway
1.23 Wireless Technologies: Wi-Fi and Bluetooth

Unit 2: Physical and Data Link Layer

3.7 Physical Layer
3.7.6 Importance of Transmission Media: Coaxial Cable, Twisted Pair Cable, Optical Fiber, Bandwidth characteristics and comparisons
3.7.7 Purpose of Connectors: Registered Jack (RJ)-45, Straight Tip (ST), Subscriber Connector (SC), Lucent Connector (LC)
3.7.8 Explain Data Encoding & Multiplexing Techniques
3.7.8.1 Time Division Multiplexing (TDM)
3.7.8.2 Frequency Division Multiplexing (FDM)
3.7.9 Explain various Switching Techniques
3.7.9.1 Circuit Switching
3.7.9.2 Packet Switching
3.7.9.3 Message Switching

3.8 Data Link Layer

3.8.6 Define the terms: Errors, Types of Errors, Redundancy

3.8.7 Explain Coding and Block Coding,

3.8.8 Error Detection and Correction: Detection versus Correction, Forward Error Correction versus Retransmission, Parity Bits, Cyclic Redundancy Check.

3.8.9 Explain Medium Access Control (MAC) Protocols: CSMA/CD (used in Ethernet), CSMA/CA (used in Wi-Fi)

3.8.10 MAC Addressing: Format and Role in Frame Delivery

Unit 3: Network and Transport Layer Concepts

2.3 Network Layer

2.3.1 Introduction to Network Addressing.

2.3.2 List and describe the Components of IP Address.

2.3.3 List and explain IP Address Classes.

2.3.4 Define subnet and describe the necessity of sub-netting.

2.3.5 Illustrate sub-netting and Subnet mask calculations.

2.3.6 Explain sub-netting with a simple example

2.3.7 List the advantages and disadvantages of sub netting

2.3.8 Describe the Internet Protocol Addressing: IPv4 and IPv6

2.3.9 State the need for IPv6.

2.3.10 Explain about Classful addressing and classless addressing in IPv4.

2.3.11 Describe Internet protocol version-6 (IPv6) addressing.

2.3.12 Routing: Static vs. Dynamic Routing

2.3.13 List the functions of Router.

2.4 Transport Layer

2.4.1 Transport Layer Services

2.4.2 Addressing: Port Numbers

2.4.3 Connection establishment (3-way handshake) and connection release

2.4.4 Flow Control, Error Control – Sliding window.

2.4.5 Explain the features of UDP

2.4.6 TCP and UDP: differences

Unit 4: Session, Presentation & Application Layers Concepts

4.1 Explain the functions of Session Layer

4.1.1 Role of Session Layer in the OSI model

4.1.2 Session establishment, maintenance, and termination

4.2 Explain the functions of Presentation Layer

4.2.1 Role of Presentation Layer in the OSI model

4.2.2 Data formatting, encryption/decryption-Private Key & Public Key

4.2.3 Cryptography and compression

4.3 Application Layer Protocols: Purpose and working principles

4.3.1 HTTP – Hypertext Transfer Protocol

4.3.2 FTP – File Transfer Protocol

4.3.3 SMTP – Simple Mail Transfer Protocol

4.3.4 DNS – Domain Name System

4.3.5 Telnet – Remote login protocol

4.4 Address Resolution & Configuration

4.4.1 DHCP – Dynamic Host Configuration Protocol (Dynamic IP assignment)

4.4.2 ARP / RARP – Address Resolution Protocol and Reverse ARP (IP-to-MAC and MAC-to-IP mapping)

Unit 5: Network Security and Emerging Concepts

5.1 Explain about Network Security

5.1.1 Define basic security principles (CIA triad)

5.1.2 Firewalls: types, configuration

5.1.3 Intrusion Detection Systems (IDS) – basic idea

5.2 Overview of Emerging Concepts

5.2.1 Discuss the principles and uses of Virtual Private Networks (VPNs).

5.2.2 Explain the key features and applications of Delay Tolerant Networks (DTN).

5.2.3 Explore the role of Sensor Networks in intelligent applications

5.2.4 Describe Vehicular Networks, and their role in intelligent transport systems.

5.2.5 Explain the fundamentals and features of 5G Communication Technology.

CO-PO/PSO Matrix:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM403.1	3	2	2	1	1	-	2	2	3	-
CM403.2	3	3	3	-	-	-	2	2	3	-
CM403.3	3	3	3	1	1	-	2	3	3	2
CM403.4	2	2	2	2	-	-	2	2	2	2
CM403.5	2	2	3	1	2	2	3	3	2	3
Average	**2.6**	**2.4**	**2.6**	**1**	**0.8**	**0.4**	**2.2**	**2.4**	**2.6**	**1.4**

3 = strongly mapped, 2=moderately mapped, 1=slightly mapped

5 Method is to relate the level of PO with the number of hours devoted to the COs which address the given PO.

6 If > 40% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 3.

7 If 25 to 40% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 2.

8 If 5 to 25% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 1.
9 If < 5% of classroom sessions addressing a particular PO, it is considered that PO is not addressed.

COURSE CONTENT:

Unit 1: Fundamentals of Networking and Data communication: Basics of Data Communication: Components– Data representation– Data flow– Analog and Digital Signals–Periodic and Non-Periodic Signals– Sine Wave characteristics– Digital Signal concepts –Transmission Impairment –Transmission Modes – Parallel and Serial Transmission–Asynchronous and Synchronous Transmission

Need for Networking – Purpose and benefits –Types of Networks – LAN, MAN, WAN, PAN –Network Standards and Models – OSI and TCP/IP –OSI Model – Layers and functions –TCP/IP Model – Layers and functions –OSI vs. TCP/IP – Comparison – Network Topologies – Bus, Ring, Star, Mesh, Hybrid (advantages/ disadvantages) – Network Devices – Repeater, Hub, Switch, Router, NIC, Modem, Gateway –Wireless Technologies – Wi-Fi, Bluetooth

Unit 2: Physical and Data Link Layer:

Physical Layer: Transmission Media – Coaxial, Twisted Pair, Optical Fiber – Media Comparison – Bandwidth, speed, interference –Connectors – RJ-45, ST, SC, LC – Multiplexing – TDM, FDM, bandwidth sharing – Switching Methods – Circuit, Packet, Message

Data Link Layer: Error Concepts – Types of errors, redundancy –Coding – Block coding –Error Control – Parity, CRC, FEC vs. retransmission –MAC Protocols – CSMA/CD (Ethernet), CSMA/CA (Wi-Fi) – MAC Addressing – Format and use

Unit 3: Network and Transport Layer Concepts

Network Layer: IP Addressing – Structure, network & host ID –IP Classes – A, B, C, D, E– Subnetting – Concept, benefits, calculations – Internet Protocols – IPv4 and IPv6 –Addressing Methods – Classful, Classless (IPv4), IPv6 format – Routing – Static and Dynamic – Router – Functions.

Transport Layer: Services – Addressing: Port Numbers – Connection establishment (3-way handshake) and connection release- Flow Control, Error Control – Sliding window – Features of UDP– TCP and UDP: differences

Unit 4: Session, Presentation & Application Layers Concepts: Session Layer: Role – Starting, managing, ending sessions – Presentation Layer: Functions – Data format, encryption /decryption– Private & public key –cryptography and compression – Application Layer Protocols: HTTP, FTP, SMTP, DNS, Telnet – Purpose and working – Address Resolution & Configuration: DHCP – Dynamic IP assignment – ARP / RARP – IP-to-MAC and MAC-to-IP mapping

Unit 5: Network Security and Emerging Concepts: Wireless Networks- Network Security-CIA Triad – Firewalls – Intrusion Detection basics – Emerging Network Concepts: Virtual Private Networks – principles and uses – Delay Tolerant Networks – features and applications – Vehicular Networks –role in intelligent transport systems – Sensor Networks – Fundamentals of 5G.

TEXT BOOKS:

5 **Computer Networks - Andrew S. Tanenbaum & David J. Wetherall,** Pearson Education, 5th Edition, ISBN: 978-0132126953

6 **Data Communications and Networking -** Behrouz A Forouzan, Tata McGraw-Hill, 5th edition, ISBN: 9780070634145

7 **Basics of Networking,** PHI learning Pvt. Ltd. 2013, ISBN: 978-81-203-2489

REFERENCE BOOKS:

6 **Computer Networking: A Top-Down Approach, Kurose and Ross,** Pearson Education, 7th Edition, ISBN: 978-0133594140

7 **A First Course in Computer Networking, Ullman and Widom,** Vikas Publishing, ISBN: 978-0138876470

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TESTS

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 3.1
Unit test-2	From 3.2 to 5.2

26CM-404E Cyber Security

Course code	Course Title	No. of Periods/Weeks	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM404E	**Cyber Security**	**3**	**45**	**30**	**70**	**2**

Time Schedule

S.No.	Chapter/Unit Title	No. of Periods	Marks	No of Short Answer Questions	No of Assay type Questions	CO's Mapped
1.	**Introduction to Cyber Security**	14	29	3	2.5	CO1
2.	**Cyberspace and the law & Cyber forensics**	13	29	3	2.5	CO2
3.	**Cybercrime in Mobile and Wireless Devices**	9	18	2	1.5	CO3
4	**Privacy Issues Basic Data Privacy Concepts**	9	18	2	1.5	CO4
Total		45	94	10	8	

Course Objectives

i. To understand fundamental concepts, principles, and terminologies in cyber security.

ii. To recognize and analyze various types of cyber threats, and attacks

iii. To understand cyber laws, regulatory frameworks, and ethical considerations in cyberspace.

iv. To gain knowledge of cyber forensics processes, tools, and techniques.

v. To understand security issues in mobile, wireless, and IoT devices.

Course Outcomes

At the end of the course, the student shall be able to		
CO1	**CM-404.1**	To identify fundamental cyber security concepts and explain various types of cyber-attacks and cyber-crimes
CO2	**CM-404.2**	To interpret various cyber laws, regulations, and policies, and explain the role of cyber forensics in investigation.
CO3	**CM-404.3**	To assess threats to mobile, and wireless devices and recommend appropriate security measures.
CO4	**CM-404.4**	To analyze privacy issues in different domains, interpret privacy policies, and recommend protection strategies.

Learning Outcomes:

1.0 Introduction to Cyber Security

1.1 Define cyber security and explain its fundamental principles.

1.2 List and explain different layers of security.

1.3 Classify and list various types of Cyber Attacks

1.4 Explain different web-based attacks such as phishing, SQL injection, and cross-site scripting.

1.5 Explain system-based attacks such as malware, ransomware, and DoS/DDoS.

1.6 Define and differentiate between vulnerability, threat, and harmful acts with examples.

1.7 Describe internet Governance – Challenges and Constraints

1.8 Define Computer Criminals and CIA Triad (Confidentiality, Integrity, Availability)

1.9 Identify the motives of attackers (financial, political, hacktivism, espionage).

1.10 Explain types of Attacks.

1.10.1 passive Attacks

1.10.2 Software Attack

1.10.3 Hardware Attacks

1.11 What is Security Policy and explain different policies.

2.0 Cyberspace and the law & Cyber forensics

2.1 Define cyberspace and explain its characteristics.

2.2 Describe global and national cyber security regulations.

2.3 Discuss NIST compliance and its relevance to cyber security standards.

2.4 Explain National Cyber Security Policy of India – objectives, strategies, and limitations.

2.5 Define the term Cyber Forensics and Digital Forensics

2.6 Explain the need for computer forensics in cybercrime investigation.

2.7 Explain different approaches for analysis of email Forensics

2.8 Discuss different tools of email forensics.

2.9 Describe life cycle of digital Forensics

2.10 List different types of Forensics Investigations

2.11 Discuss different challenges faced in computer forensic investigations.

3.0 Cybercrime in Mobile and Wireless Devices

3.1 List different Mobile and Wireless devices.

3.2 Describe current trends in mobility and connected devices.

3.3 Discuss Security Challenges specific to mobile devices, such as OS vulnerabilities and unsecured communication.

3.4 Describe Registry and configuration Settings for Mobile Devices

3.5 Describe different types of attacks on mobile/cell phones (SIM cloning, malware, network spoofing, SMS phishing).

3.6 Describe Organizational mobile security Policies.

4.0 Privacy Issues: Basic Data Privacy Concepts

4.1 Explain various types of data privacy attacks such as identity theft and surveillance

4.2 Discuss Data linking and profiling with examples.

4.3 Explain various Privacy policies and their specifications

4.4 Describe privacy policy languages and their applications.

4.5 Discuss privacy requirements in various domains such as healthcare, finance, e-commerce, and social media.

CO-PO / PSO Matrix:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM-404.1	3	3	2	2	2	1	1	3	2	2
CM-404.2	2	2	2	2	1	3	2	2	2	1
CM-404.3	2	3	3	3	2	1	1	3	2	3
CM-404.4	2	2	3	2	2	3	2	2	2	3
Average	2.25	2.5	2.5	2.25	1.75	2	1.5	2.5	2	2.25

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

- Method is to relate the level of PO with the number of hours devoted to the COs which address the given PO.
- If > 40% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 3
- If 25 to 40% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 2
- If 5 to 25% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 1
- If < 5% of classroom sessions addressing a particular PO, it is considered that PO is not-addressed.

COURSE CONTENT

1. Introduction to Cyber Security: Basic Cyber Security Concepts-layers of security-Vulnerability- threat- Harmful acts- Internet Governance – Challenges and Constraint- Computer Criminals-CIA Triad-Assets and Threat-motive of attackers- active attacks-passive attacks-Software attacks-hardware attacks-Cyber Threats-Cyber Warfare-Cyber Crime- Cyber terrorism-Comprehensive Cyber Security Policy.

2. Cyberspace and the Law & Cyber Forensics: Introduction- Cyber Security Regulations-Roles of International Law-The INDIAN Cyberspace- National Cyber Security Policy-Introduction to Cyber Forensics-Digital Forensics Science-The Need for Computer Forensics-Cyber Forensics and Digital evidence- Forensics Analysis of Email- Digital Forensics Lifecycle-Forensics Investigation-Challenges in Computer Forensics

3.Cybercrime in Mobile and Wireless Devices: Introduction- Proliferation of Mobile and Wireless Devices-Trends in Mobility-Credit card Frauds in Mobile and Wireless Computing Era-Security Challenges Posed by Mobile Devices- Registry Settings for Mobile Devices-Authentication service Security-Attacks on Mobile/Cell Phones-Organizational security Policies and Measures in Mobile Computing Era.

4.**Privacy Issues: Basic Data Privacy Concepts**: Fundamental Concepts-Data Privacy Attacks-Data linking and profiling-privacy policies and their specifications- privacy policy languages-privacy in different domains- medical, financial, e-commerce etc

Table specifying the scope of syllabus to be covered for unit tests

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 2.5
Unit test-2	From 2.6 to 4.5

Course code	Course Title	No. of Periods / Week	Total No. of periods	Marks for FA	Marks for SA	CREDITS
26CM405E	**CLOUD COMPUTING**	3	45	30	70	**2**

TIME SCHEDULE

S.No.	Chapter/Unit Title	No. of Periods	Weightage of Marks	No. of Short Answer Questions	No of Essay Questions	CO's Mapped
1.	Introduction to Cloud Computing	**8**	**14**	**2**	**1**	**CO1**
2.	Parallel and Distributed Computing Concepts	**10**	**22**	**2**	**2**	**CO2**
3.	Virtualization and Cloud Architecture Services	**14**	**33**	**3**	**3**	**CO3**
4.	Cloud Deployment Models and Cloud Economics	**13**	**25**	**3**	**2**	**CO4**
Total		**45**	**94**	**10**	**8**	

COURSE OBJECTIVES:

Upon completion of the course the student shall be able to:	
(i)	Learn what cloud computing is and the basics of related types like cluster, grid, parallel, and distributed computing.
(ii)	Understand what virtualization means, and describe how cloud services are organized (IaaS, PaaS, SaaS) and what the main parts of cloud systems are.
(iii)	Know about different types of cloud setups (public, private, hybrid, community) and get to know the main companies that offer cloud services
(iv)	Discuss about how using cloud computing affects costs, what problems or benefits it might bring, and what new developments—like better security or easier scaling—are happening in the cloud.

COURSE OUTCOMES:

CO1	CM405.1	Describe what cloud computing is and understand key related paradigms such as cluster, grid, parallel, and distributed computing.
CO2	CM405.2	Explain the role of virtualization in cloud computing, and clearly describe the main cloud service models — Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).
CO3	CM405.3	Recognize different cloud deployment models like public, private, hybrid, and community clouds, and identify major cloud service providers and their offerings.
CO4	CM405.4	Analyse the cost factors, challenges, benefits, and latest developments in cloud computing including security, scalability, and emerging technologies.

LEARNING OUTCOMES

1.0: Introduction to Cloud Computing

1.1 Define cloud computing, including basic terminology.
1.2 Describe the **history and evolution of cloud computing**.
1.3 List the **core features** and **principles** of cloud computing.
1.4 Define and distinguish recent computing trends:
- 1.4.1 Cluster Computing
- 1.4.2 Grid Computing
- 1.4.3 Distributed Computing
- 1.4.4 Utility Computing

1.5 Identify the major **cloud service providers** currently in the market.
1.6 Enumerate **challenges** in cloud computing.
1.7 Summarize the **advantages and disadvantages** of cloud computing.

2.0: Parallel and Distributed Computing Concepts

2.1 Trace the main **eras of computing** leading up to cloud computing.
2.2 Explain **parallel computing**:
- 2.2.1 Define parallel computing and its key concepts.
- 2.2.2 Describe hardware architecture for parallel processing.
- 2.2.3 Explain approaches and levels of parallelism.

2.3 Understand **distributed computing**:
- 2.3.1 Define distributed computing and its general concepts.
- 2.3.2 List components and architectural styles (software and system) for distributed systems.
- 2.3.3 Explain models for inter-process communication.

2.4 Identify technologies for distributed computing:
- 2.4.1 Remote Procedure Call (RPC)
- 2.4.2 Distributed Object Framework
- 2.4.3 Service-Oriented Computing

3.0: Virtualization and Cloud Architecture

3.1 Define **virtualization** and explain its importance.
3.2 Enumerate the characteristics of virtualization.
3.3 Classify and explain major virtualization techniques:
  3.3.1 Machine Reference Model
  3.3.2 Hardware Level Virtualization and its techniques
  3.3.3 OS Level Virtualization
  3.3.4 Programming Language Level Virtualization
  3.3.5 Application Level Virtualization
3.4 Discuss the role of virtualization in enabling cloud computing.
3.5 Analyze the pros and cons of virtualization for organizations and users.
3.6 Recognize popular virtualization technologies:
  3.6.1 Xen
  3.6.2 VMware
  3.6.3 Microsoft Hyper-V
3.7 Illustrate the working of cloud computing architectures.
3.8 List and define the main cloud service models:
  3.8.1 Infrastructure as a Service (IaaS): Description and characteristics
  3.8.2 Platform as a Service (PaaS): Description and characteristics
  3.8.3 Software as a Service (SaaS): Description and characteristics
3.9 Compare and contrast IaaS, PaaS, and SaaS based on features and use cases.

4.0 Cloud Deployment Models and Cloud Economics

4.1 List and define **types of cloud deployment models.**
4.2 State the purposes and use cases for each model.
4.3 Deep dive into:
  4.3.1 Public Cloud: Definition, working, pros, and cons
  4.3.2 Private Cloud: Definition, working, pros, and cons
  4.3.3 Hybrid Cloud: Definition, working, pros, and cons
  4.3.4 Community Cloud: Definition, working, pros, and cons
4.4 Discuss the **economics of cloud computing:**
  4.4.1 Cost considerations
  4.4.2 Operational versus capital expenditure
  4.4.3 Scalability and business agility benefits
4.5 Appreciate ongoing and emerging trends impacting cloud computing
  4.5.1 Edge Computing, Serverless Computing
  4.5.2 AI and Machine Learning in the Cloud
  4.5.3 Cloud Security Innovations
  4.5.4 Emerging Service Models (e.g., Function as a Service, Container Services)

CO-PO-PSO MAPPING

Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	0	0	1	0	1
CO2	3	1	2	2	1	0	0
CO3	2	1	2	1	2	1	1
CO4	1	3	1	0	2	2	2
Average	**2.25**	**1.75**	**1.67**	**1**	**1.5**	**1.5**	**1.33**

COURSE CONTENT:

1. Introduction to Cloud Computing

Definition and Evolution, **Key Features and Principles**, Essential Characteristics Basic principles behind cloud operation, **Recent Computing Paradigms**, Cluster Computing, Grid Computing, Distributed Computing, Utility Computing, **Cloud Ecosystem**, Overview of Cloud Service Providers, Challenges in Cloud Computing, Advantages and Disadvantages of Cloud Computing

2. Parallel and Distributed Computing Concepts

Evolution of Computing, Eras of Computing: Mainframe, Client-Server, Internet, Cloud, **Parallel Computing**, Hardware Architecture for Parallel Processing, Approaches, Pipeline Parallelism, Levels of Parallelism, **Distributed Computing**, Components of Distributed Systems, Software Architectural Styles: Client-Server, n-tier, Peer-to-peer, System Architectural Styles: Centralized, Decentralized, Hybrid, Inter-process Communication, Distributed Technologies: RPC, Distributed Object Framework, Service Oriented Architecture (SOA)

3. Virtualization and Cloud Architecture Services

Basics of Virtualization, Characteristics of Virtualization, Types of Virtualization, Virtualization in Cloud Computing, Role and Importance in Cloud Infrastructure, **Pros and Cons, Examples of Virtualization Technologies,** Xen, VMware, Microsoft Hyper-V, **Cloud Architecture, Cloud Service Models,** Infrastructure as a Service (IaaS)-Features, Providers, Use Cases, Platform as a Service (PaaS)-Features, Providers, Use Cases, Software as a Service (SaaS)-Features, Providers, Use Cases, **Comparison of Service Models**

4. Cloud Deployment Models and Cloud Economics

Overview of Deployment Models, Public Cloud, Private Cloud, Hybrid Cloud Community Cloud, Selection and Economics, Purposes and Selection Criteria, Introduction to Cloud Economics, Economics of Cloud Computing, Cost Considerations: CapEx vs. OpEx, Return on Investment (ROI), Scalability and Business Agility, Edge Computing, Serverless Computing, AI and Machine Learning in the Cloud, Cloud Security Innovations, Emerging Service Models

REFERENCE BOOKS:

1. Cloud computing a practical approach - Anthony T.Velte , Toby J. Velte Robert Elsenpeter, TATA McGraw- Hill , New Delhi – 2010
2. Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online - Michael Miller - Que 2008
3. Cloud computing for dummies- Judith Hurwitz , Robin Bloor , Marcia Kaufman ,Fern Halper, Wiley Publishing, Inc, 2010
4. Cloud Computing (Principles and Paradigms), Edited by Rajkumar Buyya, James Broberg, Andrzej Goscinski, John Wiley & Sons, Inc. 2011
5.

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TEST-1 & UNIT TEST-2

Unit Test	Learning outcomes to be covered
Unit test-1	1.1 to 3.3
Unit test-2	3.4 to 4.5

TROUBLESHOOTING OF COMPUTER NETWORKS

Course code	Course Title	No. of Periods / Week	Total No.of periods	Marks forFA	Marks for SA	Credits
26CM406A	**TROUBLESHOOTING OF COMPUTER NETWORKS**	**2**	**30**	**NIL**	**NIL**	**NIL**

Time Schedule

Unit No.	Unit/Chapter Title	No. of Periods	COs Mapped
Unit 1	Basics of Troubleshooting and Network Faults	10	CO1
Unit 2	Troubleshooting Tools and Commands	10	CO2
Unit 3	Troubleshooting Wired and Wireless Networks	10	CO3
Total		**30**	

COURSE OBJECTVES

Upon successful completion of this course, the student will be able to:

1. Identify and analyze network-related issues in LAN/WAN environments.
2. Understand error types in hardware and software components of networks.
3. Use diagnostic tools and commands to resolve common network problems.

COURSE OUTCOMES(Cos)

At the end of the course, the student shall able to		
CO1	AU1.1	Identify common network faults and failures.
CO2	AU1.2	Use hardware and software tools for network troubleshooting.
CO3	AU1.3	Troubleshoot connectivity issues in wired and wireless networks.

CO-PO/PSO Matrix:

CONO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM406.1	3	3	2	2	1	2	2	3	2	2
CM406.2	3	3	3	2	2	2	2	3	2	2
CM406.3	3	3	3	3	2	2	2	3	2	2
Average	**3**	**3**	**2.67**	**2.33**	**1.67**	**2**	**2**	**3**	**3**	**2**

3=stronglymapped,2=moderatelymapped,1=slightlymapped

- Method is to relate the level of PO with the number of hours devoted to the COs which address the given PO.
- If > 40% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 3
- If 25 to 40% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 2
- If 5 to 25% of classroom sessions addressing a particular PO, it is considered that PO is addressed at Level 1
- If < 5% of classroom sessions addressing a particular PO, it is considered that PO is not-addressed.

Learning Outcomes:

Unit 1: Basics of Troubleshooting and Network Faults

1.1 Fundamentals of troubleshooting?
- 1.1.1 What is trouble shooting?
- 1.1.2 Types of Maintenance: Preventive & Corrective Maintenance
- 1.1.3 Various approaches to Troubleshooting
  - 1.1.3.1 Top-Down Approach
  - 1.1.3.2 Bottom-up Approach
  - 1.1.3.3 Divide and Conquer
  - 1.1.3.4 Follow the Path
  - 1.1.3.5 Spot the difference
- 1.1.4 Key Steps in Network Troubleshooting

1.2 Common network faults:
Cable issues, NIC failure, switch/router faults

1.3 Types of errors:
Link down, IP conflict, DNS resolution error

1.4 Importance of logs and reporting

1.5 Service Desk Process:
Basic ticket lifecycle – Ticket creation, classification, assignment, resolution, closure

Unit 2: Troubleshooting Tools and Commands

2.1 OS-level tools:
ping, tracert / traceroute, ipconfig / ifconfig, netstat, nslookup, arp, route

2.2 Graphical tools:
Wireshark, Advanced IP Scanner, Cisco Packet Tracer

2.3 Hardware tools:
Cable tester, Loopback plug, Crimping tools

2.4 Hands-on activities:
- 2.4.1. Crimp RJ-45 connectors
- 2.4.2. Test Ethernet cables using LAN cable tester
- 2.4.3. Observe and interpret LED indicators on switches and routers

Unit 3: Troubleshooting Wired and Wireless Networks

3.1 Wired LAN issues:

3.1.1 Bad cables
3.1.2 Loose connectors
3.1.3 Improper crimping

3.2 LAN setup:
3.2.1 Assign static/dynamic Ips
3.2.2 verify subnet mask and gateway
3.2.3 test network connectivity
3.2.4 Addressing mis configurations (e.g., IP conflicts, subnet errors, VLAN issues)

3.3 Diagnosing Common Network Faults
3.3.1 Loss of network connectivity
3.3.2 slow network performance
3.3.3 intermittent disconnections

3.4 Packet analysis: Use Wireshark to capture and examine packets (e.g., ARP, ICMP, DNS)

3.5 Wireless network problems:
3.5.1. Interference from other devices/networks
3.5.2. Weak signal coverage/ coverage gaps
3.5.3. Authentication failures (wrong password, encryption type mismatch)

3.6 Wireless troubleshooting activities:
3.5.4. Change SSID and security settings
3.5.5. Analyze signal strength
3.5.6. Resolve connection issues by adjusting placement, channel, or encryption

Lab Activities (Suggestive)

Unit 1: Basics of Troubleshooting and Network Faults

1. **Exercise 1.1:** Identify Physical Layer Faults
   - **Activities:** Unplug NIC, use faulty cable, disable network interface
   - **Objective:** Detect and isolate hardware faults
   - **Tools**: Ethernet cable, PC NIC settings
2. **Exercise 1.2:** Simulate and Troubleshoot IP Address Conflict
   - **Activities:** Assign same IP to two hosts in a LAN and resolve the conflict
   - **Objective:** Understand IP address management and conflict resolution
   - **Tools**: PCs, LAN Switch
3. **Exercise 1.3:** Diagnose and Resolve DNS Resolution Error
   - **Activities:** Use incorrect DNS entry and troubleshoot using nslookup
   - **Objective:** Understand name resolution
   - **Tools**: Command prompt / Terminal, DNS server or public DNS (e.g., 8.8.8.8)
4. **Exercise 1.4:** Simulate and Troubleshoot Link Down Scenarios
   - **Activities**: Disconnect cable or disable port and restore connectivity
   - **Objective:** Practice identifying and resolving Layer 1/2 link issues and resolution

- **Tools**: Switch/router, cable

5. **Exercise1.5:** Apply Structured Troubleshooting and Documentation
   - **Activities:** Document a network issue and follow the ticket lifecycle
     Simulate a network issue (e.g., no internet)
     Use step by step troubleshooting process
     Fill out a troubleshooting report/log
   - **Objective:** Use a structured troubleshooting method to diagnose and resolve a simulated issue
   - **Tool:** Excel or simulated ticketing software (osTicket), windows/Linux system

6. **Exercise 1.6:** Log File Analysis for Network Issues
   - **Activities**: Use Windows Event Viewer or Linux log files to trace network events
   - **Objective:** Learn how to interpret logs for fault diagnosis
   - **Tools**: Windows/Linux system

Unit 2: Troubleshooting Tools and Commands

7. **Exercise 2.1:** Use Using OS-level Network Commands for Network Testing
   - **Activities**: Test IP connectivity using ping, View routing table and ARP cache, Resolve DNS using nslookup
   - **Objective:** Understand diagnostic command outputs
   - **Tools:** ping, tracert, ipconfig, netstat, nslookup, arp, route
8. **Exercise 2.2:** Capture and Analyze Packets Using Wireshark
   - **Activities:** Capture packets during web access or ping, filter DNS/HTTP/ARP
   - **Objective**: Observe packet-level communication and identify errors
   - **Tools**: Wireshark
9. **Exercise 2.3:** Discover devices and open ports in a subnet Using IP Scanning Tools
   - **Activities**: Scan subnet with Advanced IP Scanner; identify live hosts and open ports
   - **Objective**: Practice IP Scanning and Host Discovery
   - **Tools:** Advanced IP Scanner
10. **Exercise 2.4:** Crimp RJ-45 Cables and Test
    - **Activities**: Create straight-through/crossover cables; test using LAN cable tester
    - **Objective:** Verify cable quality and pinout
    - **Tools**: Crimping tool, cable tester, RJ-45 connectors, Cat5e/Cat6 cables
11. **Exercise 2.5:** Use Loopback Plugs and Cable Testers
    - **Activities:** Check NIC functionality and test cable continuity
    - **Objective:** Understand use of loopback plugs; Validate hardware and cabling with tools
    - **Tools**: Loopback plug, LAN cable tester

Unit 3: Troubleshooting Wired and Wireless Networks

12. **Exercise 3.1:** Configure and Troubleshoot a Wired LAN
    - **Activities**: Assign static and dynamic IP addresses
      Set subnet mask and gateway
      Use ping to test connectivity
    - **Objective:** Perform LAN setup and diagnose IP/subnet/VLAN misconfigurations
    - **Tools**: PCs, Switch, DHCP server/router
13. **Exercise 3.2:** Diagnose Wired Network Performance
    - **Activities**: Capture traffic in problematic scenarios (e.g., high latency)
    - **Objective:** Identify retransmissions, delays, and dropped packets
    - **Tools**: Wireshark, Switch, PCs
14. **Exercise 3.3:** Set Up and Troubleshoot Wireless Networks
    - **Activities:** Change SSID/security settings
      Measure signal strength and adjust channel
      Simulate and resolve auth errors
    - **Objective:** Resolve coverage, interference, and auth errors
    - **Tools**: Wireless router, Wi-Fi analyzer app, laptop
15. **Exercise 3.4:** Simulate Network Faults Using Packet Tracer
    - **Activities:** Create a small topology in Cisco Packet Tracer or GNS3
      Introduce faults (e.g., wrong IP, missing cable, switch off)
      Diagnose and resolve
    - **Objective:** Practice fault handling virtually
    - **Tools:** Cisco Packet Tracer

Required Software:

Software	Purpose	Platform
Wireshark	Packet capture and analysis	Windows/Linux
Cisco Packet Tracer	Network simulation	Windows/Linux
Advanced IP Scanner	Host discovery & scanning	Windows
Event Viewer	View system and network logs	Windows
Notepad++ / Excel	Logging and ticket simulation	Windows/Linux
Basic ticketing software (e.g., osTicket)	To manage customer support requests	Windows/Linux

Required Hardware and Tools:

Hardware / Tools	Purpose
PCs/Laptops (Min 2)	Testing, configuration, and monitoring of network setups
RJ-45 Cables	Provides physical LAN connectivity
Crimping Tool	Used to attach RJ-45 connectors to Ethernet cables

LAN Cable Tester	Tests continuity and pin configuration of Ethernet cables
Loopback Plug / Adapter	Tests NIC and serial port functionality
Network Switch	Facilitates basic wired LAN connectivity between devices
Wi-Fi Router/Access Point	Provides wireless network setup and troubleshooting
Multimeter (optional)	Measures voltage, continuity in electrical components/cables
Cable Stripper	Strips insulation from cables before crimping
Tone Generator	Traces and identifies cables within a bundle or wall
Cables (Cat5e/Cat6)	Used to build or replace Ethernet connections
Labeling Stickers / Tags	For labeling cables and ports for easier identification

Suggested Textbooks & References

1. **"Computer Networks"** by Andrew S. Tanenbaum
2. **"Networking All-in-One for Dummies"** by Doug Lowe
3. **Cisco Networking Academy** Course Materials
4. Official Manuals for Wireshark, Packet Tracer, Windows/Linux networking commands

Recommended Resources

1. Cisco Networking Academy – *Network Troubleshooting Module*
2. CompTIA Network+ – *Troubleshooting Section*
3. LiveAction, FITA, GeeksforGeeks – Articles on troubleshooting techniques
4. Wireshark & Packet Tracer official documentation

Course code	Course Title	No. of Periods /Weeks	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM407L	**Web Technologies Lab**	**6**	**90**	**40**	**60**	**2.5**

TIME SCHEDULE:

S.No.	Chapter/Unit Title	No. of Periods	WEIGHTAGE OF MARKS	CO's Mapped
1.	HTML & Basic Web Design	15	12	CO1
2.	CSS & XML	18	18	CO2, CO3
3.	JavaScript & JQuery	21	25	CO1, CO2
4.	PHP	21	25	CO3, CO4
5.	Mini – Project	15	10	CO1, CO2, CO3, CO4, CO5
	Total Periods	90	90	

COURSE OBJECTIVES:

i) Understand the fundamental principles of designing clear, effective, and user-friendly web pages.
ii) Learn to structure and style web content using HTML and CSS.
iii) Develop the skills to create functional websites that incorporate both client-side and server-side components.
iv) Gain familiarity with essential web technologies such as JavaScript, jQuery, and PHP.
v) Understand how to connect web applications to a database using PHP for dynamic content management.

COURSE OUTCOMES

CO1	CM407.1	**Design and implement interactive web pages using HTML, CSS, and JavaScript.**
CO2	CM407..2	**Demonstrate the usage and features of jQuery to enhance user interaction and DOM manipulation.**
CO3	CM407.3	**Develop dynamic web applications using server-side programming with PHP.**
CO4	CM407.4	**Implement database connectivity using PHP and MySQL to manage and retrieve data.**
CO5	CM407.5	**Create real-world web applications using a combination of front-end and back-end tools and technologies.**

CO-PO/PSO MATRIX:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM407.1	**2**	**2**	**3**	**2**	**-**	**2**	**1**	**2**	**3**	**2**
CM407.2	**1**	**3**	**3**	**3**	**1**	**3**	**1**	**3**	**3**	**3**
CM407.3	**-**	**2**	**3**	**2**	**1**	**3**	**1**	**2**	**3**	**3**
CM407.4	**1**	**1**	**3**	**2**	**2**	**3**	**2**	**2**	**3**	**3**
CM407.5	**3**	**3**	**3**	**3**	**2**	**3**	**2**	**3**	**3**	**3**
Average	**1.5**	**2.6**	**3**	**2.6**	**1.5**	**3**	**1.5**	**2**	**3**	**1.5**

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

8 Method is to relate the level of PO with the number of hours devoted to the COs which address the given PO.
9 If > 40% of lab sessions addressing a particular PO, it is considered that PO is addressed at Level 3.
10 If 25 to 40% of lab sessions addressing a particular PO, it is considered that PO is addressed at Level 2.
11 If 5 to 25% of lab sessions addressing a particular PO, it is considered that PO is addressed at Level 1.
12 If < 5% of lab sessions addressing a particular PO, it is considered that PO is not addressed.

LIST OF EXERCISES:

HTML & Basic Web Design

8 Create a Simple Web Page with Basic Tags using <html>, <head>, <title>, <body>, and text formatting tags.

9 Design a Web Page Using Lists and Character Entities using ordered, unordered, and definition lists along with character entities.

10 Create a Web Page Using Tables by using <table>, <thead>, <tbody>, <tfoot>, <th>, <td>, rowspan, colspan.

11 Build a Web Page with Links and Images using <a> and <img> with various attributes (target, href, src, alt, etc.).

12 Create a Web Form for User Registration using <form>, <input>, <textarea>, <select>, <option>, <fieldset>, <legend>.

CSS

13 Apply Inline, Internal, and External CSS to a Web Page to demonstrate styling using different CSS levels.

14 Use CSS Selectors and Box Model Properties

Style elements using class, ID selectors; apply padding, margin, border.

15 Design a Responsive Layout Using Flexbox and Grid

Use display: flex and display: grid to build two-column layouts.

XML

16 Create a Simple XML Document - Define a custom XML structure (e.g., student list) and ensure it follows rules.

JavaScript & AngularJS

17 Use basic JavaScript validation for required fields and pattern matching.

18 Create a calculator or age calculator using functions and onclick.

19 Display student details using arrays and manipulate them dynamically.

20 Show current date and time using Date, perform calculations using Math.

21 Build simple, interactive web pages using core AngularJS directives.

JQuery

22 Access elements using element, ID, and class selectors. Use .css() to change styles.

23 Demonstrate JQuery Effects using .hide(), .show(), .fadeIn(), .slideToggle() on divs.

24 Manipulate DOM Using JQuery using .html(), .text(), .append() to dynamically update content.

25 Handle Events with JQuery using click, hover, focus, submit events to trigger alert or form validation.

PHP – Basics

26 Install the following on local machine:

v) Apache Web server

vi) MySQL

vii) PHP and configure it to work with Apache Web server and MySQL.

27 Display a welcome message using echo, variables, and constants.

28 Write a program that manipulates strings and arrays (e.g., reverse a string, sort array).

29 Create a PHP Script for Form Handling - Read form data using $_POST and display it.

PHP & MySQL

30 Connect to MySQL Database and Display Data using mysqli_connect, fetch data from a table, and display using PHP.

31 Insert Form Data into MySQL Database - Create a form, insert submitted data into a MySQL table.

32 Update and Delete Records in MySQL using PHP Create pages to edit and delete records using UPDATE and DELETE.

PHP – Cookies & Sessions

33 Write a script to create a cookie and display its value on another page.

34 Create a Login Page Using Sessions Authenticate user, start session, and display a welcome message on a protected page.

Note: Mini project – Few examples as listed below.

1. Student Registration System

Build a complete student registration form that saves data to a MySQL database.

2. Online Feedback Form with Admin View

Collect feedback from users and display it in an admin panel.

3. Simple Product Catalogue with Search

Display a list of products pulled from a database with a search option.

4. User Login & Profile Page

Implement a basic authentication system with session handling.

5. Online Event Booking Application

Create a system to book seats for events or workshops.

****

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TEST-1 & UNIT TEST-2

Unit Test	Learning outcomes to be covered
Unit test-1	1 to 17
Unit test-2	18 to 34

COMMUNICATION AND EMPLOYABILITY SKILLS

Course code	Course Title	No. Of periods / week	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM408L	COMMUNICATION AND EMPLOYABILITY SKILLS	4	60	40	60	2

TIME SCHEDULE

S. No.	Chapter/ Unit Title	No. of Periods	CO's Mapped
1.	ABC of Communication	6	CO1
2.	Let's Learn to Listen	6	CO2
3.	I am...	4	CO4
4.	Let's Talk About...	4	CO4
5.	JAM	6	CO4
6	Interpreting Data	6	CO3
7	Your Perfect Profile	4	CO5
8	Group Discussion	8	CO4
9	Interview Skills	8	CO5
10	Making Presentations	8	CO3
	Total	**60**	

COURSE OBJECTIVES

(i)	To impart verbal and non-verbal communication skills
(ii)	To foster employability skills among the students for career building

COURSE OUTCOMES:

Upon completion of the course, the student shall be able to:

CO1	CM408.1	Practise appropriate body language and etiquette
CO2	CM408.2	Listen and comprehend the listening inputs related to different genres effectively
CO3	CM408.3	Interpret data and give oral and written presentations in academic and professional contexts
CO4	CM408.4	Communicate effectively in interpersonal interactions, interviews, and group discussions
CO5	CM408.5	Exhibit employability skills: job hunting, resume writing, attending interviews

LEARNING OUTCOMES

UNIT 1: ABC of Communication

1.1. Identify and practice the process of communication.

1.2. Demonstrate befitting body language traits to enhance communication.

1.3. Apply appropriate strategies to minimize various barriers of communication.

1.4. Communicate effectively in a given situation with a purpose.

UNIT 2: **Let's Learn to Listen**

2.1. Identify and distinguish different phonic sounds in English language.

2.2. Practice active listening in the process of communication.

2.3. Comprehend diverse listening inputs in academic, professional and everyday situation using appropriate strategies.

UNIT 3: I am...

3.1. Prepare an organized self-introduction for formal and informal situations.

3.2. Introduce yourself in job interviews effectively.

3.3. Use appropriate body language while introducing yourself.

UNIT 4: Let's Talk About...

4.1. Describe objects, places, events and people using appropriate adjectives.

4.2. Use appropriate sentences and expressions while describing anything.

4.3. Use suitable adjectives to convey mood or tone.

UNIT 5: JAM

5.1. Generate ideas on a given topic.

5.2. Organize the ideas sequentially for an effective JAM speech.

5.3. Speak spontaneously and fluently on a given topic within the stipulated time.

UNIT 6: Data Interpretation

6.1. Identify different forms of graphs, charts, diagrams and tables.

6.2. Analyse and interpret data.

6.3. Present the inferences and findings in spoken and written communication.

UNIT 7: Your Perfect Profile

7.1. Draft a customised professional resume.

7.2. Create a professional Applicant Tracking System (ATS) compliant Resume.

7.3. Draft a cover letter to communicate with prospective employers.

UNIT 8: Group Discussion

8.1. Identify the significance of group discussion and differentiate the various stages involved.

8.2. Practice various roles and skills involved in group discussion.

8.3. Demonstrate appropriate body language for effective participation in group discussion.

UNIT 9: Interview Skills

9.1. Practice proper interview demeanour.

9.2. Respond effectively to frequently asked interview questions (FAQs).

9.3. Demonstrate readiness to job opportunities.

UNIT 10: Making Presentations

10.1. Demonstrate the features of a good presentation.

10.2. Use appropriate presentational aids.

10.3. Prepare and give presentations on various topics effectively.

CO-PO/PSO Mapping

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO6	PO7	PS01	PS02	PS03
CO1	PO's 1 to 5 are not directly applicable for English course.					1	1	Programme Specific Outcomes are Branch specific with technical aspects which are not directly applicable to English Language course.		
CO2						1	1			
CO3						2	2			
CO4						2	2			
CO5						2	2			
Average						1.6	1.6			

3-Strongly Mapped 2- Moderately Mapped 1- Slightly Mapped

Note: The gaps in CO and PO mapping will be met by one or more appropriate activities from the following:
(i) Assignments (ii) Tutorials (iii) Seminars (iv) Guest Lectures (v) Group Discussions (vi) Quizzes (iv) Library Visits etc.,

REFERENCE BOOKS

A Textbook of English Phonetics for Indian Students by T. Balasubramaian

Better English Pronunciation by J.D. O'Connor

Group Discussion for Admissions and Jobs by Anand. S. Ganguly

Communicative English by E. Suresh Kumar and P. Sreehari

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR INTERNAL ASSESSMENTS I & II

Assessment	Learning Outcomes to be Covered
Internal Assessment – I	From 1.1 to 5.3
Internal Assessment – II	From 6.1 to. 10.3

Course code	Course Title	No. of Periods /WeeK	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM409L	**JAVA PROGRAMMING LAB**	**6**	**90**	**40**	**60**	**2.5**

TIME SCHEDULE:

S.No.	Chapter/Unit Title	No. of Periods	Weightage of Marks	CO's Mapped
1.	Basics, overloading, inheritance, overriding	14	15	CO1,CO2
2.	Streams, Interfaces and Packages and Collections.	12	10	CO2,CO3
3.	Exceptions and Multi threaded programming.	14	15	CO3, CO4
4.	Swings and Event Handling	20	20	CO5
	Total	60	60	

COURSE OBJECTIVES:

i)Design object-oriented programming paradigm

ii)Able to develop multi-tasking application with the knowledge of multi-threading.

iii) Familiarized to develop graphical user interface with event handling mechanism

COURSE OUTCOMES:

CO1	CM409.1	Perform object-oriented programming model application design.
CO2	CM409.2	Design optimized definition for an application with reusability features like inheritance and polymorphism.
CO3	CM409.3	Analyze modular design for real time applications by using packages concept in projects. Able to design data structures used in applications.
CO4	CM409.4	Apply multi threading concepts to implement multitasking

		and multi programming applications.
CO5	CM409.5	Develop effective dynamic user interface for any front end applications using Swings and events.

CO-PO/PSO MATRIX:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM409.1	2	1	3	2		2	1	2	3	3
CM409.2	1	3	3	3	1	3	2	2	3	3
CM409.3	1	2	3	2	2	3	1	2	3	3
CM409.4	1	1	3	2	2	3	2	2	3	3
CM409.5	3	3	3	3	2	3	2	2	3	3
Average	1.5	2.6	3	2.6	1.5	3	1.6	2	3	3

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

- Method is to relate the level of PO with the number of hours devoted to the COs which address the given PO.
- If > 40% of lab sessions addressing a particular PO, it is considered that PO is addressed at Level 3.
- If 25 to 40% of lab sessions addressing a particular PO, it is considered that PO is addressed at Level 2.
- If 5 to 25% of lab sessions addressing a particular PO, it is considered that PO is addressed at Level 1.
- If < 5% of lab sessions addressing a particular PO, it is considered that PO is not addressed

LEARNING OUTCOMES

1. Exercise programs on conditional statements and loop statements
2. Exercise programs on Strings.
3. Exercise program to create class and objects and adding methods.
4. Exercise programs using constructors and construction over loading.
5. Exercise programs on command line arguments.
   i) Input as command line arguments and perform operation on that data.
   ii) Input as command line arguments and update manipulated data in Files.
6. Exercise programs using concept of overloading methods.
7. Exercise programs on inheritance.
8. Write a program using the concept of method overriding.
9. Exercise on packages.

i) Creation of packages
ii) Design module to importing packages from other packages.

10. Exercise programs on interfaces.
11. Exercise programs on I/O Streams
    i) Reading data through Keyboard
    ii) Perform Reading and Writing operations on files using File Streams.
12. Exercise programs on Collections.
    i) Write a java program to search a student mark percentage based on pin number using Array list.
    ii) Write a java program to create linked list to perform delete, insert, and update data in linked list with any application.
13. Exercise on exception handling.
    i) Programs on try, catch and finally.
    ii) Programs on multiple catch statements
    iii) Programs on nested try statements.
14. Exercise on multithreading
    i) Programs on creation of single and multiple threads.
    ii) Programs on adding priorities to multiple threads.
15. Exercise on applets
    i) Programs on Graphics and colors.
    ii) Simple animations using threads and graphics.
16. Exercise on AWT controls and Swings
    i) Program to handle mouse events.
    ii) Program to handle keyboard events.
    iii) Programs to illustrate JText Fields andJ Button control.
    iv) Programs to illustrate JCheckBox and JList control.
    v) Write an application program to illustrate multiple controls.
    vi) Programs to illustrate JFrame and JDialog with controls.

KEY COMPETENCIES

Exp. No.	Name of the experiment	Objectives	Key Competencies
1	Exercise programs on conditional statements and loop statements.	(a) Write program using if statement and switch (b) Write program using while, do and for constructs.	(a) Know the usage of IF and switch statements. (b) Compile the program and rectify the errors. (c) Observe the output.
2	Exercise programs on Strings.	(a) Write a programs to manipulate Strings (b) Write a programs to arrange array of strings in ascending order	(a) Create String objects (b) Use string class methods Observe the output.
3	Exercise program to create class and objects and adding methods.	(a) Write a program to create a class and create objects. (b) Write a program to create class adding methods and access class members.	(a) Create class. (b) Declare methods. (c) Create objects. (d) Write main method. (e) Access class members.
4	Exercise programs using constructors and construction over loading.	(a) Write a program using default constructor. (b) Write a program using parameterized constructor.	(a) Declare and define constructor. (b) Call default constructor. (c) Call parameterized constructor. (d) observe constructor overloading.
5	Exercise programs on command line arguments.	(a) Write a program to illustrate usage of command line arguments. (b) Write a program to read data as command line arguments and update it into Files.	(a) Use command line arguments. (b) Run the program. (c) Understand usage of Files. (c) Observe the output.
6	Exercise programs using concept of overloading methods.	(a) Write a program to illustrate method overloading. (b) Write a program to illustrate method overloading using constructors.	(a) Observe method overloading. (b) Overload constructor methods.

7	Exercise on inheritance.	(a)Write a program to illustrate single inheritance. (b)Write a program to illustrate multiple inheritance.	(a) Create base class. (b) Write base class constructor. (c) Create derived class. (d) Use extends keyword. (e) Use super keyword. (f) Write derived class constructor.
8	Write a program using the concept of method overriding.	Write a program using the concept of method overriding.	(a) Use method overriding. (b) Use this keyword. (c) use super keyword
9	Exercise on importing packages.	Write a program to create and importing package.	(a) Create package. (b) Use of access specifiers. (b) Use package. (c) Use import keyword.
10	Exercise on interfaces.	Write a program to illustrate multiple inheritance using interfaces.	(a) Define interface. (b) Use extends keyword. (c) Use implements keyword. (d) Access interface variables.
11	Exercise programs on I/O Streams	(a) Write a program to give values to variables interactively through the keyboard. (b) Write program to read and write primitive data types. (c) Write programs to handle Files.	(a) Use different data types. (b) Use readLine() method. (c) Use println() method. (d)use File Streams Observe the output.
12	Exercise programs on Collections.	(a) Write a java program to search a student mark percentage based on pin number using Array list. (b)Write a java program to create linked list to perform delete, insert, and update data in linked list with any application.	(a) Define collection classes (b) use ArrayList, LinkedList (c) apply List and Iterator Interface
13	Exercise on exception handling	(a) Write a program to illustrate exception handling. (b) Write a program to	(a) Use try – catch. (b) Use multiple catch blocks.

		illustrate exception handling using multiple catch statements. (c) Write a program to illustrate exception handling using nested try.	(c) Use finally statement. (d) use Nested try
14	Exercise on multithreading	(a) Write a program to create single a thread by extending the thread class. (b) Write a program to create a single thread by implementing the runnable interface. (c) Write a program to create multiple threads. (d) Write a program to illustrate thread priorities.	(a) Use extends, new. (b) Use run() and start() methods. (c) Observe thread execution. (d) Use implements runnable interface. (e) Use setPriority() and getPriority() methods. (f) use wait(),notify() methods
15	Exercise on Swings.	Write a program to create simple applet to display different shapes with colors. Write an applet program to design simple animation.	(a) Use <applet>…</applet> tag. (b) Add applet to html file. (c) Run the applet. (d) use graphics methods (e) use threads and graphics.
16	Exercise on AWT controls using swings controls	(a) Write an applet program to handle key events. (b) Write an applet program to handle mouse events. (c) Write a swing program to illustrate JTextField and JButton control. (d) Write a swing program to illustrate JCheck box and JList control. (e)Write swing program to illustrate multiple controls.	(a) Use keyboard event methods (b) Use mouse event methods (c) Use JText Field class methods (d) Use JButton class methods (e) Use JCheckbox and JList class methods

FOR INTERNAL ASSESSMENTS I & II

Assessment	Learning Outcomes to be Covered
Internal Assessment – I	From 1 to 8
Internal Assessment – II	From 9 to 16

FIFTH SEMESTER								
THEORY SUBJECTS	COMPULSORY SUBJECTS	SNo	**Sub. Code**	Subject	PRACTICUM	Periods	Total Periods	Credits
		1	26CM501T	BASICS OF QUANTUM COMPUTING	N	6	90	4
		2	26CM502T	SOFTWARE ENGINEERING	N	6	90	4
		3	26CM503T	ANDROID PROGRAMMING	N	6	90	4
	ELECTIVE SUBJECTS	ELECTIVE SUBJECT						
		4	26CM504E	IoT	N	3	45	2
		5	26CM505E	FUNDAMENTALS OF AI & ML	N			
	AUDIT SUBJECT							
		6	26CM506A	UI / UX DESIGN	Y	2	30	
		TOTAL THEORY + PRACTICUM PERIODS				23	345	14
PRACTICAL SUBJECTS	COMPULSORY SUBJECTS	7	26CM507L	ANDROID PROGRAMMING LAB	Y	4	60	1.5
		8	26CM508L	AI USING PYTHON	Y	6	90	2
		9	26CM509P	PROJECT WORK	N	6	90	2
				ACTIVITY	…	3	45	0.5
		TOTAL PRACTICAL + PRACTICUM PERIODS				19	285	6
GRAND TOTAL PERIODS (15 WEEKS PER SEMESTER)						42	630	20

BASICS OF QUANTUM COMPUTING

Course code	Course Title	No. of Periods/Weeks	Total No. of periods	Marks for FA	Marks for SA	Credits
C26CM501T	**BASICS OF QUANTUM COMPUTING**	**6**	**90**	**30**	**70**	**4**

TIME SCHEDULE:

S.No.	Chapter/Unit Title	No. of Periods	Marks	No. of Short Answer Questions	No. of Essay Type Questions	CO's Mapped
1.	Introduction to Quantum Computing	18	14	2	2	CO1
2.	Qubits and Superposition	18	22	2	2	CO2
3.	Quantum Gates and Circuits	20	22	2	1	CO3
4.	Entanglement and Quantum Teleportation	19	14	2	1	CO3,CO4
5.	Quantum Computing in AI & ML	15	22	2	2	CO3,CO5
Total Periods		90	94	10	8	

COURSE OBJECTIVES:

- Understand the fundamental differences between classical and quantum computing concepts.
- Learn the evolution, history, and key terminology of quantum computing.
- Comprehend the principles of qubits, superposition, and quantum measurement.
- Gain conceptual knowledge of quantum gates, circuits, and their role in quantum computation.

- Explore the phenomena of quantum entanglement and teleportation and their applications.
- Recognize how quantum computing can enhance Artificial Intelligence and Machine Learning techniques.
- Analyze real-world applications, hardware platforms, and current research trends in quantum technology.
- Develop foundational understanding required for further studies or practical work in quantum computing and AI/ML integration.

COURSE OUTCOMES:

At the end of the course the student able to learn following:		
CO1	CM501.1	**Describe the basic principles and evolution of quantum computing**, distinguishing it from classical computing systems.
CO2	CM502.1	**Explain the concept of qubits, superposition, and quantum states**, and understand their role in quantum computation.
CO3	CM503.1	**Construct and interpret simple quantum circuits using quantum logic gates**, and simulate quantum operations using basic tools or visualizations.
CO4	CM504.1	**Understand and apply the principles of entanglement and quantum teleportation**, and discuss their implications in secure data transmission.
CO5	CM505.1	**Identify and explain potential applications of quantum computing in AI and ML**, and evaluate how quantum algorithms may accelerate machine learning tasks.

LEARNING OUTCOMES:

Chapter 1 – Introduction to Quantum Computing

1.1 Introduction to classical vs quantum computing.
1.2 History and evolution of quantum computing.
1.3 Key differences between bits and qubits (conceptual view).
1.4 Limitations of classical computing.
1.5 Quantum computing advantages and challenges.
1.6 Quantum computing terminology (conceptual glossary).
1.7 Real-world applications overview.
1.8 Major milestones in quantum computing.
1.9 Overview of quantum hardware platforms (IBM, Google, etc.).
1.10 Future prospects and trends in quantum technology.

Chapter 2 – Qubits and Superposition

2.1 Concept of a qubit.
2.2 Physical representation of qubits (photons, electrons, etc.).
2.3 Difference between 0/1 bits and qubits.
2.4 Superposition explained using analogies.
2.5 Measurement and its effect on qubits.
2.6 Quantum state examples without equations.
2.7 Quantum decoherence (basic understanding).
2.7 Examples of superposition in daily life.
2.9 Role of superposition in computational speed-up.

Chapter 3 – Quantum Gates and Circuits

3.1 Introduction to quantum logic gates.
3.2 Single-qubit gates (X, Y, Z, H) – conceptual only.
3.3 Multi-qubit gates (CNOT, SWAP) – conceptual only.
3.4 Gate operations as transformations without math.
3.5 Building a simple quantum circuit (conceptual).
3.6 Difference between classical and quantum logic gates.
3.7 Quantum parallelism (idea only).
3.8 Circuit representation diagrams (without equations).
3.9 Practical examples of gate usage.

Chapter 4 – Entanglement and Quantum Teleportation

4.1 Concept of quantum entanglement.
4.2 Real-world analogies for entanglement.
4.3 Applications of entanglement in communication.
4.4 Role of entanglement in secure communication.
4.5 Quantum key distribution concept.
4.6 Entanglement in quantum networks.
4.7 Experimental demonstrations of teleportation.
4.8 Limitations and challenges of entanglement.

Chapter 5 – Quantum Computing in AI & ML

5.1 Introduction to AI & ML.
5.2 Why quantum computing can enhance AI & ML.
5.3 Examples of quantum-enhanced machine learning.
5.4 Concept of quantum data encoding (no math).
5.5 Quantum pattern recognition (conceptual).
5.6 Quantum clustering – high-level explanation.
5.7 Industries using quantum AI.
5.8 Case studies of quantum ML applications.
5.9 Challenges and future research directions.

CO-PO/PSO MATRIX:

CO NO.	PO1	PO 2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM501.1	1	1	1	1	0	0	2	0	1	1
CM501.2	1	0	1	2	1	0	2	2	2	0
CM501.3	3	0	0	1	0	0	2	2	1	0
CM501.4	2	2	1	1	0	0	1	2	1	3
CM501.5	2	0	1	1	1	0	2	2	3	0
Average	1.8	0.6	0.8	1.2	0.5	0	1.8	1.6	1.6	0.8

3=Strongly mapped , 2=moderately mapped, 1=slightly mapped

COURSE CONTENT:

Chapter 1 – Introduction to Quantum Computing:

Introduction to classical vs quantum computing-History and evolution of quantum computing-Key differences between bits and qubits (conceptual view)-Limitations of classical computing-Quantum computing advantages and challenges-Quantum computing terminology (conceptual glossary)-Real-world applications overview-Major milestones in quantum computing- Overview of quantum hardware platforms (IBM, Google, etc.)- Future prospects and trends in quantum technology.

Chapter 2 – Qubits and Superposition:

Concept of a qubit-Physical representation of qubits (photons, electrons, etc.)-Difference between 0/1 bits and qubits-Superposition explained using analogies-Measurement and its effect on qubits-Quantum state examples without equations-Quantum decoherence (basic understanding)-Examples of superposition in daily life-Role of superposition in computational speed-up.

Chapter 3 – Quantum Gates and Circuits:

Introduction to quantum logic gates-Single-qubit gates (X, Y, Z, H) – conceptual only.
Multi-qubit gates (CNOT, SWAP) – conceptual only-Gate operations as transformations without math-Building a simple quantum circuit (conceptual)-Difference between classical and quantum logic gates-Quantum parallelism (idea only)-Circuit representation diagrams (without equations)-Practical examples of gate usage

Chapter 4 – Entanglement and Quantum Teleportation

Concept of quantum entanglement-Real-world analogies for entanglement-Applications of entanglement in communication-Role of entanglement in secure communication.
Quantum key distribution concept-Entanglement in quantum networks. Experimental demonstrations of teleportation-Limitations and challenges of entanglement.

Chapter 5 – Quantum Computing in AI & ML

Introduction to AI & ML-Why quantum computing can enhance AI & ML. Examples of quantum-enhanced machine learning-Concept of quantum data encoding (no math)-Quantum pattern recognition (conceptual)-Quantum clustering – high-level explanation-Industries using quantum AI-Case studies of quantum ML applications.Challenges and future research directions.

REFERENCE BOOKS:

1) Quantum Computation and Quantum Information
Authors: Michael A. Nielsen & Isaac L. Chuang Publisher: Cambridge University Press

2) Dancing with Qubits: How quantum computing works and how it can change the world
Author: Robert S. Sutor Publisher: Packt Publishing / IBM Press

3) Quantum Computing: An Applied Approach

4) Learn Quantum Computing with Python and Q#
Authors: Sarah Kaiser, Christopher Granade, Nathan Wiebe, and Brian C. Beckman Publisher: Manning Publications

5) Quantum Machine Learning: What Quantum Computing Means to Data Mining Author: Peter Wittek

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TESTS

Unit Test	Learning outcomes to be covered
Unit test-1	From 1.1 to 3.5
Unit test-2	From 3.6 to 5.9

Course code	Course Title	No. of Periods/ Weeks	Total No. of periods	Marks for FA	Marks for SA
26CM502T	Software Engineering	6	90	30	70

Time Schedule

Chapter No.	Chapter/Unit Title	No. of Periods	Marks	No. of Short Answer Questions	No. of Essay Type Questions	CO's Mapped
1.	Basics of Software Engineering Designs & Life Cycle Models	13	14	2	1	CO1
2.	Software Project Management	21	22	2	2	CO2
3.	Requirement Analysis & Specifications	13	11	1	1	CO3
4.	Software Design, Coding	23	25	3	2	CO4
5.	Software testing, Debugging, Reliability, Quality Management & Maintenance	20	22	2	2	CO5
Total		**90**	94	10	8	

Course Objectives

Course Objectives	i) To apply the fundamental concepts of software engineering & software development lifecycle (SDLC) models to real-world development scenarios. ii) To apply basic principles of software project management including planning, scheduling and risk analysis. iii) To design software solutions using software engineering principles and UML diagrams. iv) To apply software testing techniques to evaluate reliability and performance.

Course Outcomes

At the end of the course the student shall be able to :		
CO1	CM-502.1	Apply the fundamentals of Software Engineering to differentiate between various software life cycle models.
CO2	CM-502.2	Apply Software Project Management principles to plan, estimate, schedule and manage risks
CO3	CM-502.3	Develop a SRS document by identifying and organizing functional and non-functional requirements for a given software problem
CO4	CM-502.4	Apply software design principles, UML modeling, and coding standards to develop maintainable software.
CO5	CM-502.5	Apply software testing techniques and quality metrics to evaluate software reliability and performance.

Learning Outcomes:

1.0 Basics of Software Engineering Designs & Life Cycle Models

1.1 Know the Evolution and Impact of the Software Engineering
  1.1.1 Know the evolution of software engineering as an Art to an Engineering Discipline
  1.1.2 Know how software engineering is a solution to the Software Crisis?
1.2 Know the difference between Programs and Software Products
1.3 Understand the evolution of Software Engineering Design
  1.3.1 Know about Early Computer Programming
  1.3.2 Know about High Level Language Programming
  1.3.3 What is control Flow-Based design
  1.3.4 What is Data Structure oriented design
  1.3.5 What is Data Flow-oriented design
  1.3.6 What is Object Oriented design
  1.3.7 Know about Model Driven Architecture
  1.3.8 Know about DevOps based Design
  1.3.9 Know about Low-code/No-code Development
1.4 Explain the Software Life Cycle Models
  1.4.1 Explain Classical Waterfall Model
  1.4.2 Explain Iterative Waterfall Model
  1.4.3 Explain Prototyping Model
  1.4.4 Explain Evolutionary Model
  1.4.5 Explain Spiral Model
  1.4.6 Know about AGILE Development Model- principles and Life Cycle
  1.4.7 Compare different Life Cycle Models
1.5 Explain Software Engineering ethics and the responsibilities of software professionals

2.0 Software Project Management

2.1 Know about Software Project Manager
  2.1.1 List the Job Responsibilities of a Software Project Manager
  2.1.2 Identify skills necessary for Software Management
2.2 Explain Software Project Planning
2.3 Explain the components of SPMP document
2.4 Know the Metrics for Project Size Estimation
  2.4.1 Describe Lines of Code (LOC)
  2.4.2 Describe Function Point Metric
2.5 Know about Project Estimation Techniques
  2.5.1 What is Empirical Estimation Technique
  2.5.2 What is Heuristic Estimation Technique
2.6 List Staffing Level Estimates
2.7 Understand the Basics of Agile Project Management using SCRUM
2.8 Know about Scheduling
  2.8.1 Explain Work Break Down Structure
  2.8.2 Explain Activity Networks

2.8.3 Explain Gantt Charts
2.8.4 Explain PERT Charts
2.9 Explain the process of Staffing in a Software Project Management
2.10 List the Attributes of a Successful Software Engineer
2.11 Explain the Risk Mnagement
2.11.1 What is Risk Identification
2.11.2 Know about Risk Assesment
2.11.3 Know about Risk Containment

3.0 Requirement Analysis and Specifications

3.1 Explain Requirements Gathering and Analysis
3.2 Explain Software Requirement Specifications
3.2.1 Describe the Purpose and importance of SRS
3.2.2 Summarize IEEE 830 standard
3.3 Explain Functional Requirements and Use Cases
3.3.1 Define Functional Requirements
3.3.2 Describe procedure to Identify requirements using scenario-based use cases
3.3.3 Explain how to document functional requirements
3.4 Explain requirements Traceability and Construct a Traceability matrix
3.5 List Characteristics of a Good SRS Document
3.6 Give Examples of Poor SRS Document
3.7 Explain the typical Organization of the SRS Document
3.8 Develop an SRS for an ATM or Library System

4.0 Principles of Software Design, Coding

4.1 Know the Software Design Concepts
4.1.1 List the characteristics of good software design
4.1.2 What is Cohesion and Coupling
4.1.2.1 Define Cohesion and Coupling
4.1.2.2 Classify and compare types of Cohesion
4.1.2.3 Classify and compare types of Coupling
4.2 Know about Software Design Approaches
4.2.1 Explain Function-Oriented Design
4.2.2 Explain Object-Oriented Design
4.2.3 Compare Function-Oriented and Object-Oriented Design
4.2.4 Describe Service-Oriented Design
4.3 Understand User Interface Design
4.3.1 List the Characteristics of a good User Interface
4.3.2 Understand the Basic Concepts of user interface design
4.3.2.1 Describe User Guidance and Online Help
4.3.2.1 Differentiate Mode Based vs Modeless Interface
4.3.2.2 Compare Graphical vs Text-Based User Interface
4.3.3 Know the Types of User Interface
4.3.3.1 Describe Command Language Based Interface
4.3.3.2 Describe Menu Based Interface

4.3.3.3 Describe Direct Manipulation Interfaces

4.4 Understand Unified Modeling Language

4.4.1 Define UML and its purpose in Software Engineering

4.4.2 Explain the importance of UML in Object oriented Design

4.4.3 List and Explain Building blocks of UML

4.4.4 List different symbols used in UML notation

4.4.5 Classify and list standard UML diagrams

4.4.6 Know about Class Diagrams

4.4.6.1 Describe the purpose of Class Diagram

4.4.6.2 Know how to Draw simple Class Diagrams

4.4.7 Know about Use case diagram

4.4.7.1 Define Use Case and Actors

4.4.7.2 Describe the purpose of Use case diagram

4.4.7.3 Know how to Draw Use case diagrams

4.4.8 Understand Interaction diagrams

4.4.8.1 Explain the purpose and use of Interaction diagrams

4.4.8.2 Distinguish between Sequence and Communication Diagrams

4.4.8.3 Know how to Draw Interaction diagrams for System Scenarios

4.5 Know the Concept of Software Coding

4.5.1 Explain Coding Standards

4.5.2 Discuss Coding guidelines

4.5.3 Explain the process of Code Review

4.5.4 Describe Code Walkthroughs

4.5.5 Explain Code Inspection

4.5.6 Define Clean Room Testing

4.5.7 Explain the importance of Software Documentation

5.0 Testing, Debugging, Reliability, Quality Management & Maintenance

5.1 Understand Testing

5.1.1 Define Software Testing and its purpose

5.1.2 Differentiate Verification and Validation

5.1.3 List and Describe common Test Case Design techniques

5.1.4 Compare Testing in the Large and Testing in the Small

5.1.5 Explain Unit Testing with suitable examples

5.1.6 Explain Black box Testing techniques

5.1.7 Explain White Box Testing techniques

5.1.8 List various Software Testing tools

5.1.9 Describe the Role of Testing in Agile and DevOps environment

5.2 Know about Debugging

5.2.1 Explain Debugging Approaches

5.2.2 List standard Debugging Guidelines

5.3 Explain Program Analysis Tools

5.3.1 Know about Static Analysis Tools

5.3.2 Know about Dynamic Analysis Tools

5.4 Explain the purpose and process of Integration Testing

5.5 Apply System Testing techniques to check the overall working of the software

5.6 Apply Performance Testing to check how fast and stable the software runs

5.7 Explain the role of Artificial Intelligence in Software Testing

5.8 Understand the basics of Object-Oriented Testing and its techniques

5.9 Understand the concept of Software Reliability

5.9.1 Differentiate Hardware Reliability and Software Reliability

5.9.2 List common reliability metrics: MTTF, MTTR, and availability

5.9.3 Identify the key steps in reliability growth modeling

5.10 Explain Software Quality Management systems

5.11 Explain SEI Capability Maturity Model (CMM) and its levels

CO-PO/PSO Matrix

CONO.	PO 1	PO 2	PO 3	PO 4	PO5	PO 6	PO 7	PSO 1	PSO 2	PSO 3
CM-502.1	3	2	2	1	1	1	1	2	2	1
CM-502.2	2	3	2	1	1	2	-	1	2	1
CM-502.3	2	3	3	2	-	1	-	1	2	2
CM-502.4	2	2	3	3	1	2	-	2	2	1
CM-502.5	2	2	2	3	1	1	1	1	2	3
Average	2.2	2.4	2.4	2	0.8	1.4	0.4	1.4	2	1.6

3=strongly mapped,2=moderately mapped,1=slightly mapped

COURSECONTENT

1. Introduction to Software Engineering-Life Cycle Models-Software Engineering Ethics
2. Software Project Management-Responsibilities of a Software Project Manager – Project Metrics- Project Estimation Techniques-Staffing Level Estimation-Agile Project Planning - Scheduling – Risk Management
3. Requirement Analysis and Specification: Requirement Gathering and Analysis-SRS document

4. Software Design , Coding : Good software design, Cohesion and Coupling, Software Design Approaches, User interface Design, Software Coding and Goals of UML - Role of UML in Object oriented Design - Building blocks of UML : Things, Relationships, and Diagrams - Symbols used in UML notation - Classify and list standard UML diagrams - Class diagram, purposes of class diagram, draw the class diagram - Use case diagram, define the term Use case, purposes of Use case diagram, draw the Use case diagram - Interaction diagram, purposes of Interaction diagram, the types of interaction diagrams : Sequence diagram and Collaboration diagram, draw the Interaction diagrams.

5. Software Testing, Debugging, Reliability, Quality Management and maintenance – Testing, Debugging software Reliability- Statistical Testing, Software Quality, Software Quality Management System, SEI capability Maturity Model.

REFERENCEBOOKS

1. Fundamentals of Software Engineering – Rajib Mall(PHI) Second Edition.
2. Software Engineering - Jawadekar(TMH)
3. **Software Engineering: A Practitioner's Approach** – *Roger S. Pressman* & *Bruce R. Maxim*, McGraw Hill (8th or 9th edition)
4. Software Engineering Concepts - Fairley(TMH)
5. An Integrated Approach to Software Engineering-*Pankaj Jalote*, Narosa Publishing House, 2nd Edition
6. **The Unified Modeling Language User Guide** – *Grady Booch, James Rumbaugh, Ivar Jacobson*
7. http://www.tutorialspoint.com/uml/ – UML concepts and examples (Free Online Resource)

Table specifying the scope of syllabus to be covered forunit tests

Unit Test	Learning outcomes to be covered
Unittest-1	From1.1to3.8
Unittest-2	From4.1to5.11

Course code	Course Title	No. of Periods/ Weeks	Total No. of periods	Marks for FA	Marks for SA	Credits
26CM503 T	**Android Programming**	**6**	**90**	**30**	**70**	**4**

Time Schedule

Chapter No.	Chapter/Unit Title	No. of Periods	Marks	No. of Short Answer Questions	No. of Essay Type Questions	CO's Mapped
1.	Android Basics and Anatomy	15	14	2	1	CO1
2.	Components, Activity Life Cycle, Intents	15	14	2	1	CO2
3.	Android – User Interface	20	22	2	2	CO3
4.	Android Advanced Concepts	20	22	2	2	CO4
5.	Data Base connectivity in Android	20	22	2	2	CO5
Total		**90**	94	10	8	

Course Objectives

Course Objectives	At the end of the course, the student shall be able to
	i)To know the Basics of Android Application Development ii)To familiarize with the Android Anatomy, Components, Activity Life Cycle, Intents iii)To use various User Interface controls in Android Application Development iv)To reinforce theoretical concepts by creating relevant Android applications.

Course Outcomes

At the end of the course, the student shall be able to	
CO1	Explain Basics of Android Programming
CO2	Comprehend the Anatomy, Components, Activity Life Cycle, Intents
CO3	Incorporate the User Interface Controls in Android Programming
CO4	Analyse the Android Advanced Concepts in Android Programming
CO5	Integrate the DataBase with Android Applications, Publish Android applications & Deploy APK files

Learning Outcomes

1.0 Android Basics, Android Anatomy

- **1.1** List the mobile phone generations
- **1.2** Define Smart phone
- **1.3** Explain the Android Architecture
- **1.4** Describe Android versions
- **1.5** List Features of Android
- **1.6** Explain installation of Android SDK Tools
- **1.7** Explain creation of Android Virtual Devices(AVD)
- **1.8** Explain Anatomy of Android Application
  - **1.8.1** Main Activity File
  - **1.8.2** Manifest File
  - **1.8.3** R file
  - **1.8.4** Strings File

1.8.5 The Layout File
1.8.6 Dalvik Virtual Machine

1.9 Explain Creation of "Hello World" Android Application.

1.10 Explain procedure for Deploying Android app on USB connected Android device

2.0 Components, Activity Life Cycle, Intents

2.1 Explain Android application components
2.1.1 Activities
2.1.2 Services
2.1.3 Content providers
2.1.4 Broadcast Receivers

2.2 Describe Activity life cycle

2.3 Define intents
2.3.1 Exploring Intent objects
2.3.2 List the Intent types
2.3.3 Linking activities using intents

2.4 Explain Fragments
2.4.1 Fragment Implementation
2.4.2 Finding Fragments
2.4.3 Adding, Removing, and Replacing Fragments
2.4.4 Finding Activity Using Fragment

2.5 Explain the creation of Android application that switches between Activities

3.0 Android – User Interface

3.1 List types of User Interface layouts
3.2 Explain User Interface Designing Layouts
3.3 Define User Interface Controls
3.4 List User Interface Controls
3.5 Describe the need of Event Handling
3.6 Explain Android Event Handling
3.7 Explain the usage of User Interface Controls
3.7.1 Textbox
3.7.2 Edit Text
3.7.3 Button
3.7.4 Toast
3.7.5 Check box
3.7.6 Radio Button
3.7.7 Toggle Button
3.7.8 Spinner
3.7.9 Date Picker
3.7.10 Time Picker
3.7.11 List View
3.7.12 Progress Bar
3.7.13 Analog and Digital clock
3.8 Explain development of Android App using above user Interface Controls

4.0 Android Advanced Concepts

- **4.1** State the purpose of Menus
- **4.2** List various types of Menus
- **4.3** Explain Menus
  - **4.3.1** Options Menus
  - **4.3.2** Context Menus
  - **4.3.3** Popup Menus
- **4.4** Explain Images
  - **4.4.1** Image Button
  - **4.4.2** Image Switcher
  - **4.4.3** Image View
- **4.5** Describe Alert Dialog.
- **4.6** Explain Alarm Manager
- **4.7** Compose the procedure for SMS sending
  - **4.7.1** List the methods in SMS Manager Class
- **4.8** Explain procedure for E-mail sending
- **4.9** Explain Media Player Class
  - **4.9.1** List various methods in Media Player class
  - **4.9.2** Explain the procedure for Running Media Player
- 4.10 Explain Telephony Manager
- 4.11 Explain the procedure of using Camera
- 4.12 Describe the procedure of Recording Video
- 4.13 Explain Home Screen Widgets
  - 4.13.1 Adding the Broadcast Receiver Class to an Android Manifest
  - 4.13.2 Using the RemoteViews and AppWidgetManager Classes
  - 4.13.3 Creating a Customized Clock Widget
- 4.14 Describe the collection View Widgets
  - 4.14.1 Collection View Widget Layouts
  - 4.14.2 Creating the Remote Views Service Class
  - 4.14.3 Creating a Remote Views Factory Interface
  - 4.14.4 Populating Collection View Widgets

5 Database connectivity in Android

- 5.1 Define persistent data storage
- 5.2 Understanding of Android storage system
  - 5.2.1 List different storage options
  - 5.2.2 Explain Internal file Storage
  - 5.2.3 Explain External file Storage
- 5.3 Know about Content Providers
  - 5.3.1 Importance of Content providers
  - 5.3.2 Explain Content Providers
- 5.4 Understand SQLite Database
  - 5.4.1 Define Schema
  - 5.4.2 Define Contract class
  - 5.4.3 List the Database Operations
  - 5.4.4 Explain SQLite DataBase
- 5.5 Explain the Database Operations with an example database
- 5.6 Explain the procedure for publishing of an Android Application
- 5.7 Explain the procedure for Deploying APK files

CO-PO/PSO Matrix

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM-503.1	3	2	1	1	1	1	2	3	2	1
CM-503.2	2	3	3	2	1	2	2	2	3	2
CM-503.3	2	2	3	3	1	2	2	2	3	2
CM-503.4	2	2	3	3	2	2	2	2	3	2
CM-503.5	2	3	3	3	2	2	3	3	3	3
Average	2.2	2.4	2.6	2.4	1.4	1.8	2.2	2.4	2.8	2

3=strongly mapped,2=moderately mapped,1=slightly mapped

COURSE CONTENT

UNIT 1:Android Basics

Types of mobile phone generations, Android Architecture, Android Versions, Features of Android, Installing Android SDK Tools, Android Development Tools (ADT), Creating Android Virtual Devices (AVD), Anatomy of android application, Deploying Android app on USB connected Android device,

UNIT 2:Android Anatomy, Components, Activity Life Cycle, Intents

Creating first android application, Android application components, Activity life cycle, understanding activities, Exploring Intent objects, Intent Types, linking activities using intents, Fragments-Fragment Implementation, Finding Fragments, Adding, Removing, and Replacing Fragments, Finding Activity Using Fragment

UNIT 3: Android – User Interface

Understanding the User Interface Layouts, User Interface Controls - Button, Toast, Toggle Button, Checkbox, Radio Button, Spinner, Edit Text, Date Picker, Time Picker, List View, Progress- Bar, Analog and Digital clock, Handling UI events, List fragment, Dialog fragment

UNIT 4: Android Advanced Concepts

Menus-Option, Context, Popup, Images- Image Button, Image Switcher, Image View, Alert Dialog, Alarm manager, SMS, E-mail, Media Player, Handling Telephony Manager, camera, Video, Widgets.

UNIT 5: Database connectivity in Android

Storing the data persistently-Data Storage Options: preferences, Internal Storage, External Storage, Content Provider, The SQLite database, connecting with SQLite

database and operations-Insert, Delete, Update, Fetch, Publishing and deploying android applications.

Reference Books:

1. Pradeep Kothari, “Android Application Development Black Book”, DreamTech Press

2. Wei-Meng Lee, Beginning Android 4 Application Development, Wiley Publishing, Inc.

3. James C.Sheusi, “Android Application Development for Java Programmers”, Cengage Learning

4. Mark L Murphy, “Beginning Android”, Wiley India Pvt Ltd

5. Sayed Y Hashimi and Satya Komatineni(2009), “Pro Android”, Wiley India Pvt Ltd

6. Reto Meier, Professional Android 4 Application Development, Wiley India Pvt Ltd Course outcomes:

7. https://developer.android.com/docs

8. https://www.geeksforgeeks.org/introduction-to-android-development/

9. https://www.tutorialspoint.com/android/index.html

Table specifying the scope of syllabus to be covered forunit tests

Unit Test	Learning outcomes to be covered
Unit test-1	1.1 to 3.8
Unit test-2	4.1 to 5.7

INTERNET OF THINGS

Course Code	Course Title	Periods/ week	Total No of Periods	Marks for FA	Marks for SA	Credits
26CM504E	**Internet of Things**	**3**	**45**	**30**	**70**	**2**

TIME SCHEDULE

Chapter No.	Chapter Title	No. of Periods	Weightage of Marks	No. Of Short Answer Questions	No. Of Long Answers Questions	Cos Mapped
1	IoT-Introduction, System Design	8	14	2	1	CO1
2	IoT Hardware and Perception Layer Protocols	12	22	2	2	CO2
3	IoT Network Layer	10	22	2	2	CO3
4	IoT Application Layer	8	22	2	2	CO4
5	Security, Privacy and Future Trends	7	14	2	1	CO5
Total		45	94	10	8	

COURSE OBJECTIVES

Upon completion of the course the student shall be able to	
i	**Grasp IoT Fundamentals**: Understand the core concepts, defining characteristics, and the layered architectural models of the Internet of Things.
ii	**Analyze Core Components**: Examine the various hardware elements like sensors and actuators, and the key communication protocols such as Bluetooth and NFC that operate at the perception layer.
iii	**Evaluate Network Protocols**: Assess and contrast network layer protocols like 6LoWPAN, RPL, and Zigbee, which are specifically designed for low-power, resource-constrained IoT environments.

iv	**Compare Application Technologies**: Differentiate between major application layer messaging protocols (e.g., MQTT, CoAP) and understand the critical role of middleware and cloud platforms in building IoT systems.
v	**Assess Advanced Topics**: Evaluate significant security vulnerabilities, address privacy and ethical challenges, and analyze the impact of emerging technologies such as AI, Block chain, and Digital Twins on the evolution of IoT.

COURSE OUTCOME

CO	CO Code	Course Outcome Statement
CO1	CM504.1	Explain the fundamental concepts, characteristics, and layered architectures of the Internet of Things.
CO2	CM504.2	Analyze the various hardware components, including sensors, actuators, and communication protocols like Bluetooth and NFC, used in the IoT perception layer.
CO3	CM504.3	Evaluate network layer protocols designed for constrained environments, such as 6LoWPAN, RPL, and Zigbee .
CO4	CM504.4	Compare different application layer messaging protocols (like MQTT and CoAP) and the role of middleware and cloud platforms in IoT systems.
CO5	CM504.5	Assess the security vulnerabilities, privacy challenges, and the impact of emerging technologies like AI and Blockchain on IoT.

LEARNING OUTCOMES

Chapter 1: IoT-Introduction, System Design

1.1. Explain the transformation of data into wisdom within an IoT system using the DIKW pyramid model.
1.2. Define the Internet of Things (IoT).
1.3. Describe the key characteristics that define an IoT system (e.g., connectivity, heterogeneity, scalability).
1.4. Recall at least six distinct application domains of IoT.
1.5. Construct and label a diagram illustrating the 4-layer IoT architecture.
1.6. Explain the functions of the additional layers (e.g., Business, Processing) in a 5-layer IoT architecture compared to a 4-layer one.
1.7. Define 'physical design' and 'logical design' in the context of IoT.
1.8. Explain the roles of 'Things,' IoT devices, and hardware interfaces in the physical design of an IoT system.
1.9. List the primary functions of an IoT gateway, such as data aggregation and protocol translation.

1.10. Describe the purpose and interaction of the main functional blocks of an IoT system, such as Device, Communication, and Services.
1.11. Categorize a list of communication protocols into their respective layers (Link, Network, Transport, Application).
1.12. Compare and contrast the Request-Response, Publish-Subscribe, Push-Pull, and Exclusive Pair communication models.
1.13. Define a communication API.
1.14. Deconstruct the operational flow of REST and Web Socket APIs, providing appropriate use cases for each in an IoT context.
1.15. Categorize different IoT systems based on their architectural complexity and functional capabilities.
1.16. Explain how key technologies like Wireless Sensor Networks (WSN), cloud computing, and big data analytics enable IoT systems.

Chapter 2: IoT Hardware and Perception Layer Protocols

2.1. State the primary function of the Perception layer in an IoT architecture.
2.2. Distinguish between a sensor and an actuator by explaining their roles in interacting with the physical environment.
2.3. Categorize various sensors based on criteria like output type, power needs, and measured quantity.
2.4. Categorize actuators based on their physical principles (e.g., hydraulic, electric) and applications.
2.5. Compare the architectures of microcontrollers and microprocessors.
2.6. Provide specific examples of IoT applications where a microcontroller (like Arduino) is a suitable choice versus those where a microprocessor (like Raspberry Pi) is more appropriate.
2.7. Describe how technologies like barcodes, QR codes, and RFID are used to identify and track devices in an IoT ecosystem.
2.8. Summarize the features, use cases, and limitations of wired technologies like Ethernet and Power Line Communication (PLC) in IoT.
2.9. Define Bluetooth.
2.10. Describe the technical specifications of Bluetooth, including its use of FHSS, network topology, and data rates.
2.11. Match specific Bluetooth profiles (e.g., A2DP, HID) to their corresponding use cases (e.g., audio streaming, input devices).
2.12. Recall major Bluetooth versions and the key advancements in each.
2.13. Define Near Field Communication (NFC).
2.14. Explain the operating principle of NFC and its different communication modes (e.g., reader/writer, peer-to-peer).
2.15. Categorize different NFC tag types and list the data formats they support.
2.16. List the advantages and common application domains for Bluetooth and NFC.

2.17. Describe the IEEE 802.15.4 standard, including its architecture, network topologies, and device types.

Chapter 3: IoT Network Layer

3.1. State the primary function of the Network layer in an IoT architecture.
3.2. Justify the need for specialized IoT protocols by explaining the limitations of standard IPv4/IPv6 for constrained devices.
3.3. Draw a diagram showing the 6LoWPAN protocol stack and its relation to other layers.
3.4. Explain the purpose of the 6LoWPAN adaptation layer and its key features like header compression and fragmentation.
3.5. Explain the purpose of the RPL routing protocol for constrained environments.
3.6. Describe the operation of RPL, including concepts like DODAGs and control messages.
3.7. Explain the synergistic relationship between 6LoWPAN and RPL.
3.8. Define Zigbee and explain why it is often preferred for low-power mesh networks.
3.9. Describe the functions of the Zigbee Network (NWK) layer and how its routing protocol works.
3.10. Recall the network topologies (star, mesh, tree) supported by Zigbee.
3.11. Recall IoT application domains where Zigbee is commonly used.
3.12. Compare and contrast the 6LoWPAN/RPL stack with the Zigbee stack, focusing on IP compatibility, flexibility, and interoperability.

Chapter 4: Application Layer

4.1. Explain how middleware facilitates IoT solutions by providing services like device management and data aggregation.
4.2. Compare cloud-based IoT platforms (like AWS IoT Core, Azure IoT Hub) with on-premise solutions based on features, services, and management models.
4.3. For brokered protocols (MQTT, AMQP) and peer-to-peer protocols (CoAP, XMPP), analyze their core principles, architectures, key features, and provide suitable IoT use cases for each.
4.4. Using a smart home scenario, illustrate how the Zigbee Application Layer, its Cluster Library (ZCL), and profiles work to enable device interoperability.

Chapter 5: Security, Privacy, and Future Trends

5.1. Identify and describe potential security vulnerabilities at the device, network, cloud, and application layers of the IoT architecture.
5.2. Propose appropriate security measures and best practices to mitigate vulnerabilities at each layer of the IoT architecture.

5.3. Explain the key privacy and ethical challenges associated with data collection in IoT systems.
5.4. Analyze the specific challenges related to obtaining user consent and protecting data in large-scale IoT deployments.-IoT Privacy
5.5. Discuss and form a judgment on the deeper moral dilemmas presented by IoT, such as algorithmic bias, surveillance, and autonomous decision-making.-IoT Ethics
5.6. Summarize the key provisions of India's DPDP Act 2023, the IT Act 2000 (in relation to CERT-In), and modern telecommunications regulations as they apply to IoT.
5.7. Explain the mechanism of integrating AI/ML with IoT and illustrate with specific applications.
5.8. Explain how block chain can be applied to IoT for enhanced security and provide relevant application examples.
5.9. Explain the concept of a Digital Twin, how it uses IoT data, and describe its impact on applications like predictive maintenance.
5.10. Discuss the dual-edged impact of quantum computing on IoT, considering both its potential benefits and the security threats it poses.

CO-PO/PSO Mapping Table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1	3	2	1	-	-	1	-	3	1	1
CO2	3	3	2	1	2	-	-	3	2	-
CO3	3	3	2	2	1	-	-	3	2	-
CO4	3	3	2	2	3	-	-	3	3	1
CO5	2	2	1	2	1	3	2	1	1	3

3 – Strong correlation (direct, significant contribution to outcome achievement), **2** – Moderate correlation (partial or indirect contribution), **1** – Low correlation (marginal contribution)

COURSE CONTENTS

Chapter-1: IoT-Introduction, Architecture

DIKW Pyramid, IoT - Definition, Characteristics, Applications, IoT System Design - Architecture, Physical Design, Logical Design, and Classification based on complexity, IoT enabled technologies.

Chapter-2: IoT Hardware and Perception Layer Protocols

Role, IoT Devices - Sensor, Actuators, End devices, Protocols - Wired (Ethernet, PLC), Wireless (IEEE 802.15.4, Bluetooth, NFC)

Chapter-3: IoT Network Layer:

6LoWPAN (purpose, functions, architecture, topologies) with RPL (purpose, functions, working, limitations) , Zigbee (NWK layer, topologies, devices, AODV, applications), Differentiate 6LoWPAN with Zigbee.

Chapter-4: IoT Application Layer

Middleware and Service Layer - Role of Middleware, Features of Cloud and On premise IoT Platforms

Application Layer Messaging Protocols - Principles, Architecture, Features and Use cases of MQTT, AMQP, CoAP, XMPP. Zigbee specific application layer protocol. Zigbee Application Layer - key components, Zigbee cluster library and profile.

Chapter-5: Security, Privacy and Future Trends

IoT Security Challenges, Solutions.

IoT privacy, ethics and regulatory frameworks with respect to India. Emerging technologies and their impact on IoT.

REFERENCE BOOKS

1. *Internet of Things: A Hands-On Approach* by Arshdeep Bahga and Vijay Madisetti.
2. *Designing the Internet of Things* by Adrian McEwen and Hakim Cassimally.
3. *IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things* by David Hanes, Gonzalo Salgueiro, Patrick Grossetete, et al.
4. *Internet of Things (IoT)* by Dr. Rajiv Chopra.
5. *The Ethical Challenges of the Internet of Things* by Adam D. Moore

UNIT TEST SYLLABUS

Unit Test 1	From chapter 1 to chapter 3.7
Unit Test 2	From chapter 3.8 to 5.10

Course Code	Course Title	Periods/ week	Total No of Periods	Marks for FA	Marks for SA	Credits
26CM505E	**Fundamentals of AI & ML**	3	45	30	70	2

TIME SCHEDULE

Chapter No.	Chapter Title	No. of Periods	Weightage	Short Answers	Long Answers	Mapped COs
1	Introduction to AI	8	11	1	1	CO1
2	Problem Solving with AI	9	22	2	2	CO2
3	Foundations of Machine Learning	10	22	2	2	CO3
4	Supervised Learning	10	25	3	2	CO4
5	Unsupervised Learning & Apps	8	14	2	1	CO5, CO6
Total		45	94	10	8	

COURSE OBJECTIVES

The primary objectives are:	
(i)	To provide a strong foundational understanding of AI principles and problem-solving techniques.
(ii)	To introduce the core concepts, types, and applications of Machine Learning.
(iii)	To equip students with the knowledge to differentiate between various AI and ML models.
(iv)	To make students aware of the real-world impact and applications of this technology, making them more industry-ready.

COURSE OUTCOMES

CO1	CM505.1	Explain the fundamental concepts of Artificial Intelligence, its history, and intelligent agents.
CO2	CM505.2	Apply various search strategies to solve well-defined problems.
CO3	CM505.3	Differentiate between supervised, unsupervised, and reinforcement learning.
CO4	CM505.4	Apply basic supervised learning algorithms for classification and regression tasks.
CO5	CM505.5	Describe the working of unsupervised learning algorithms and the basics of neural networks.
CO6	CM505.6	Identify real-world applications of AI and ML across various domains.

LEARNING OUTCOMES

1.0 *Introduction to Artificial Intelligence*

1.1 Define Artificial Intelligence and its goals
1.2 Explain the different types of AI.
1.3 Describe the structure of an intelligent agent.
1.4 Differentiate between various types of agent environments.
1.5 Analyze the structure of different agent types.

2.0 *Problem Solving with AI*

2.1 Define a problem using state space representation.
2.2 Explain the difference between uninformed and informed search.
2.3 Apply Breadth-First Search (BFS) to find a solution.
2.4.1 Apply Depth-First Search (DFS) to find a solution.
2.5 Describe how heuristics can guide a search process.

3.0 *Foundations of Machine Learning*

3.1 Define Machine Learning and its relevance.
3.2 Differentiate between AI, ML and Deep Learning.
3.3 Explain the three main types of machine learning.
3.4 Describe the typical workflow of an ML project.
3.5 Analyze the concepts of training, validation, and test datasets
3.6 Explain the concepts of overfitting and underfitting.

4.0 *Supervised Learning*

4.1 Differentiate between regression and classification problems.
4.2 Explain the working of Linear Regression for prediction.
4.3 Apply the K-Nearest Neighbors (k-NN) algorithm to classify new data points.
4.4 Explain how Decision Trees are used for classification.
4.5 Apply a simple decision tree model to solve a classification problem.
4.6 Describe basic metrics for evaluating classification models.

5.0 *Unsupervised Learning & Applications*

5.1 Explain the goal of unsupervised learning.
5.2 Apply the K-Means algorithm to group data into clusters.
5.3 Describe the basic structure of an Artificial Neural Network (ANN).

5.4 Identify various real-world applications of AI and ML.
5.5 Discuss the ethical considerations related to AI.

CO-PO/PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2
CO1	3	2	1	-	1	-	2	2	-
CO2	3	3	2	1	-	1	2	2	-
CO3	2	3	2	-	1	-	2	2	-
CO4	3	3	2	2	-	1	2	2	-
CO5	2	2	1	-	1	-	2	2	-
CO6	1	2	2	1	3	2	2	2	1
Average	2.33	2.5	1.66	1.33	1.5	1.33	2	2	1

3-Strongly Mapped 2- Moderately Mapped 1- Slightly Mapped

COURSE CONTENT

Chapter 1: Introduction to Artificial Intelligence

What is AI? A brief history, Turing Test. The AI Problems, AI Techniques, Applications of AI. Intelligent Agents: Agents and Environments. Concept of Rationality, Nature of Environments. Structure of Agents: Simple reflex agents, Model-based agents, Goal-based agents, Utility-based agents.

Chapter 2: Problem Solving with AI

Problem Solving: Define the problem as a State Space Search. Search Strategies: Uninformed (blind) vs Informed (Heuristic) Search. Uninformed Search Algorithm: Breadth-First Search (BFS), Depth-First Search (DFS), Comparing BFS and DFS on completeness, time complexity, space complexity and optimality. Informed Search: Heuristic Functions, Greedy Best-First Search, A* Search concept.

Chapter 3: Foundations of Machine Learning

Introduction to Machine Learning: What is ML? Why do we need ML? Relationship between AI, ML and Deep Learning. Types of Machine Learning: Supervised, Unsupervised and Reinforcement Learning. General ML Workflow: Data collection, preprocessing, Model Training, Evaluation, Deployment. Key Terms: Features, Labels, Training set, Test set, Validation set. Introduction to Model Evaluation: Overfitting vs Underfitting, Bias-Variance Tradeoff (conceptual)

Chapter 4: Supervised Learning

Introduction to Supervised Learning: Classification vs Regression. Regression: Simple Linear Regression. Classification: K-Nearest Neighbors(k-NN) algorithm, choosing the value of 'k'. Decision Tree Classification: Structure of a decision tree (nodes, edges, leaves), basic concept of splitting. Building a simple decision tree from a given data set. Model Evaluation for Classification: Accuracy, Confusion Matrix (True Positives, False Negatives, etc.).

Chapter 5: Unsupervised Learning & Applications

Introduction to Unsupervised Learning: The concept of finding hidden patterns in unlabeled data. Clustering: K-Means Algorithm (steps involved, choosing 'k', concept of centroids). Introduction to Neural Networks & Deep Learning: The neuron model (Perceptron), layers (input, hidden, output), activation functions. Real-world Applications of AI/ML: Image recognition, spam detection, recommendation systems (Eg. Netflix, Amazon), medical diagnosis. Ethics in AI: Bias in AI, data privacy, and the societal impact of AI.

Reference Books

1. *Artificial Intelligence: A Modern Approach* by Stuart Russell and Peter Norvig
2. *Introduction to AI & Machine Learning* by Munesh Chandra Trivedi, Ankit Srivastava
3. *Machine Learning* by Dr. S. Sridhar, and Dr. M. Vijaya Lakshmi
4. *Machine Learning* by Saikat Dutt, Subramanian Chandra Mouli, Amit Kumar Das

TABLE SPECIFYING THE SCOPE OF SYLLABUS TO BE COVERED FOR UNIT TESTS

Unit Test	Learning Outcomes to be Covered
Unit Test – 1	From 1.1 to 3.3
Unit Test – 2	From 3.4 to 5.5

Course code	Course Title	No. of Periods/Weeks	Total No. of periods	Marks for FA	Marks for SA
26CM506A	**UI / UX Design**	**2**	**30**	**00**	**00**

S.No	Chapter/Unit Title	No.of Periods	CO's Mapped
1.	Foundations of UI/UX and Human-Centered Design	10	CO1
2.	UI/UX Design Process and Prototyping Techniques	10	CO2
3.	Interactive Prototyping and Usability Testing	10	CO3
	Total Periods	30	

COURSE OBJECTIVES

i. To introduce the foundational concepts of User Interface (UI) and User Experience (UX) Design.
ii. To develop students' ability to apply the design process
iii. To enable students to create interactive prototypes and conduct usability testing

COURSE OUTCOMES

At the end of the course the student able to learn following:		
CO1	CM-506-1.1	Understand fundamental UI/UX principles and human-centered design concepts.
CO2	CM-506--1.2	Apply the UI/UX design process to develop user personas, user journeys, and wireframes.
CO3	CM-506--1.3	Develop interactive prototypes and perform basic usability testing to evaluate design effectiveness.

CO-PO/PSO Matrix:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
AU-1.1	3	2	1	1	2	1	2	3	2	1
AU-1.2	2	3	3	2	3	2	1	3	2	2
AU-1.3	1	2	3	3	3	3	2	2	3	2
Average	**2.0**	**2.3**	**2.3**	**2.0**	**2.7**	**2.0**	**1.7**	**2.7**	**2.3**	**1.7**

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

LEARNING OUTCOMES:

1.0 **Foundations of UI/UX and Human-Centered Design**.

1.1 Theory Learning Outcomes (TLOs):

1. Define UI and UX and distinguish between them
2. Describe the principles of human-centered design
3. Explain usability heuristics and accessibility guidelines
4. Understand user persona development and empathy mapping

1.2 Practical Learning Outcomes (PLOs):

1. Create a user persona using Canva or Figma
2. Perform a heuristic evaluation of a common website
3. Use Draw.io to develop an empathy map for a sample app
4. Conduct a short peer feedback session on usability

Free Tools: Figma, Canva, Draw.io, Google Forms

2.0 UI/UX Design Process – User Flows and Wireframes

2.1 Theory Learning Outcomes (TLOs):

1. List the key stages of the UI/UX design process
2. Describe how to create user flows and wireframes
3. Explain the difference between low- and high-fidelity wireframes
4. Understand the basics of information architecture and card sorting

2.2 Practical Learning Outcomes (PLOs):

1. Create a low-fidelity wireframe in Figma
2. Design a user flow using Lucidchart or Draw.io

3. Perform a card sorting activity using OptimalSort (free trial) or paper-based methods
4. Build a basic sitemap using GlooMaps

Free Tools: Figma, Draw.io, Lucidchart, GlooMaps, Miro (free for small teams)

3.0 Prototyping and Usability Testing

3.1 Theory Learning Outcomes (TLOs):

1. Explain the purpose of prototyping and usability testing
2. Describe different levels of prototypes (paper, digital, interactive)
3. Define key usability metrics
4. Understand how to collect and analyze user feedback

3.2 Practical Learning Outcomes (PLOs):

1. Create an interactive prototype using Figma's Prototyping Tool
2. Conduct a basic usability test with peers and document findings
3. Collect feedback using Google Forms and summarize user insights
4. Perform accessibility testing using WAVE browser extension

Free Tools: Figma, Google Forms, WAVE, OBS Studio (for usability recordings)

COURSE CONTENTS

UNIT – I

Foundations of UI/UX and Human-Centered Design

Introduction to UI/UX Design, Difference between UI and UX, Principles of human-centered design, Usability heuristics and Nielsen's principles, Accessibility guidelines (WCAG basics), Understanding user needs, Developing user personas, Empathy mapping techniques, Role of user feedback in UX.

UNIT – II

UI/UX Design Process – User Flows and Wireframes

UI/UX Design Process Overview, Requirement gathering and user research methods, Creating user journey maps, Defining user flows, Information

architecture and content hierarchy, Low-fidelity wireframing (sketches), High-fidelity wireframing using digital tools, Layout design principles and grid systems, Navigation design and menu structuring.

UNIT – III

Prototyping and Usability Testing

Interactive Prototyping Concepts, Tools and levels of prototype fidelity, Creating clickable prototypes using Figma, Linking wireframes and designing transitions, Planning and conducting usability tests, Collecting and analyzing user feedback, Metrics and KPIs for user experience evaluation, Introduction to accessibility tools (WAVE), Iterating based on test results..

TEXT BOOKS:

1. Don't Make Me Think – Steve Krug
2. The Design of Everyday Things – Don Norman
3. About Face: The Essentials of Interaction Design – Alan Cooper et al.

REFERENCE BOOKS:

1. Lean UX – Jeff Gothelf
2. Smashing UX Design – Jesmond Allen and James Chudley

Course code	Course Title	No. of Periods/Weeks	Total No. of periods	Marks for FA	Marks for SA	Credits
C26CM507L	**Android Programming Lab**	**4**	**60**	**40**	**60**	

TIMESCHEDULE:

S.No.	Chapter/Unit Title	No. of Periods	Weightage of marks	CO's Mapped
1.	Android Basics Activity Life Cycle	8	10	CO1
2.	Android – User Interface	16	15	CO2
3.	Android Advanced Concepts	24	20	CO3
4.	DataBase connectivity in Android	12	15	CO4
	Total Periods	60	60	

COURSE OBJECTIVES:

At the end of the course, the student shall be able to
i)To know the Basics of Android Application Development ii)To familiarize with the Android Anatomy, Components, Activity Life Cycle, Intents iii)To use various User Interface controls in Android Application Development iv)To reinforce theoretical concepts by creating relevant Android applications.

COURSE OUTCOMES

At the end of the course, the student shall be able to	
CO1	Demonstrate the Basics of Android Programming for developing Android Applications
CO2	Incorporate the User Interface Controls in Android Programming
CO3	Analyse the Android Advanced Concepts in Android Programming
CO4	Integrate the DataBase with Android Applications.

CO-PO/PSO MATRIX:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM507.1	3	2	2	3	1	2	2	3	3	2
CM507.2	2	2	3	3	1	2	2	2	3	2
CM507.3	2	3	3	3	2	2	2	2	3	2
CM507.4	2	3	3	3	2	2	3	3	3	3
Average	2.3	2.5	2.8	3.0	1.5	2.0	2.3	2.5	3.0	2.3

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

LIST OF EXCERCISES:

1. Create an Android Application To display “Hello World”
2. Create an Android app to shift from one activity to another activity using a button.
3. Create an Android app to accept a number in textfield and display the factorial of it in a Toast message on clicking a button
4. Create an Android app to illustrate the use of Datepickerwidget and Timepickerwidget.
5. Create an Android app that uses multiple UI controls like EditText, CheckBox, Spinner and Buttons.
6. Create an Android app to show Analog and Digital clock
7. Create an Android app to set an alarm.
8. Create an Android Application To Integrate Google Maps
9. Create an Android Application To send SMS
10. Create an Android Application To calling a number
11. Create an Android Application To send E-mail
12. Create an Android Application to show clock widget.
13. Create an Android Application Using Database operations like insert and display.
14. Create an Android Application Using Database operations like delete and display?
15. Create an Android Application Using Database operations like update and display?

Android Programming Lab Objectives and Key Competencies			
Sl.No	**Name of the Experiment**	**Objectives**	**Key Competencies**
1	Exercise to display "Hello World".	Create an Android app to show "Hello World"	❖ Correct syntactical errors ❖ Debug logical errors ❖ Confirm whether the required output generated properly or not
2	Exercise on Intent.0	Create an Android app to shift from one activity to another activity using a button.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Know how to apply start Activity () method using intent ❖ Confirm whether the control moves from one activity to another activity.
3	Exercise on textfield and toast message.	Create an Android app to accept a number in textfield and display the factorial of it in a Toast message on clicking a button	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the EditText and Button classes and the required methods ❖ Confirm whether the factorial is computed and shown in the Toast or not
4	Exercise on Datepicker and Timepicker	Create an Android app to illustrate the use of Datepicker and Timepicker.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the Datepicker and Timepicker class and its required methods ❖ Confirm whether the selected date and time is shown on a Toast message.
5	Exercise on multiple UIcontrols	Create an Android app that uses multiple UI controls like textfield, Checkbox, Spinner and Buttons	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the textfield, Checkbox, Spinner and Buttons classes and its required methods. ❖ Confirm whether the required operations are done properly
6	Exercise on Analog and Digital clock	Create an Android app to show Analog and Digital clock	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the Analog and Digital clock view. ❖ Confirm whether the selected time is shown as a Toast message.
7	Exercise on	Create an Android app to	❖ Correct syntactical errors ❖ Debug logical errors

Android Programming Lab Objectives and Key Competencies			
Sl.No	**Name of the Experiment**	**Objectives**	**Key Competencies**
	Alarm.	set an alarm	❖ Study the alarm class. ❖ Confirm whether the alarm is worked or not.
8	Create an Android Application To Integrate Google Maps	Create an Android Application To Integrate Google Maps	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the GoogleMap class and its methods ❖ Confirm whether the Map working properly or not
9	Create an Android Application To send SMS	Create an Android Application To send SMS	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the SmsManager class and its methods ❖ Confirm whether the messages are sending properly or not
10	Create an Android Application To calling a number	Create an Android Application To calling a number	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the class and its methods ❖ Confirm whether the event performed or not
11	Create an Android Application To send E-mail	Create an Android Application To send E-mail	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the E-mail functionality with Intents ❖ Confirm whether the E-mails are sending properly or not
12	Exercise on clock widget	Create an Android Application to show a clock widget.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the widgets ❖ Confirm whether the clock widget is properly placed or not
13	Exercise on Database operations like insert and display.	Create an Android Application Using Database operations like insert and display.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the operations of Database ❖ Confirm whether the data is properly inserted or not ❖ Confirm whether the data is properly fetched or not
14	Exercise on Database operations like delete	Create an Android Application Using Database operations like delete and display.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the operations of Database ❖ Confirm whether the data is

Android Programming Lab Objectives and Key Competencies			
Sl.No	**Name of the Experiment**	**Objectives**	**Key Competencies**
	and display.		properly deleted or not ❖ Confirm whether the data is properly fetched or not
15	Exercise on Database operations like update and display.	Create an Android Application Using Database operations like insert and display.	❖ Correct syntactical errors ❖ Debug logical errors ❖ Study the operations of Database ❖ Confirm whether the data is properly updated or not ❖ Confirm whether the data is properly fetched or not

UNIT TEST SYLLABUS

UNIT TEST-I	Learning Outcomes: 1 to 8
UNIT TEST-II	Learning Outcomes: 9 to 15

Course code	Course Title	No. of Periods / Weeks	Total No.of periods	Marks for FA	Marks for SA
26CM508L	AI using Python Lab	6	90	40	60

TIME SCHEDULE

Chapter No.	Chapter / Unit Title	No.of Periods	CO's Mapped
1.	Introduction to Python for AI & ML	18	C01
2.	Basic Machine Learning Algorithms	27	C02
3.	Fundamentals of Deep Learning using Keras/Tensor Flow	18	C03
4.	Basics of Artificial Intelligence Techniques	27	C04,C05
Total Periods		**90**	

COURSE OBJECTIVES

i. To apply core Python programming constructs for developing Artificial Intelligence and Machine Learning applications.
ii. To provide practical experience in implementing fundamental Machine Learning and Deep Learning models using Python.
iii. To analyze and solve basic AI problems using search techniques, knowledge representation, and rule-based logic implemented in Python.

COURSE OUTCOMES

CO1	Apply core Python constructs, libraries, and data structures in the context of AI and ML problem-solving.
CO2	Implement and evaluate basic machine learning algorithms using Python-based libraries such as Scikit-learn.
CO3	Build and train fundamental deep learning models using Keras/TensorFlow frameworks
CO4	Analyze and solve AI problems using uninformed and heuristic search, knowledge representation, and rule-based logic.
CO5	Apply text and image-based AI techniques such as NLP and computer vision using Python tools and libraries.

LEARNING OUTCOMES

S.No	LEARNING OUTCOME	THEORY (explained using PPT, useful videos from YouTube, etc.)	PRACTICAL (Key Competencies table enclosed)
1	Install Python, Jupyter Notebook, and essential AI/ML libraries	Overview of Python, Jupyter Notebook, and AI/ML ecosystem	Demonstration of Python, Jupyter, and library installation
2	Explore datasets using loading and inspection techniques	Explanation of Pandas data loading functions and dataset exploration	Hands-on dataset loading and inspection
3	Perform feature scaling and encoding	Data preprocessing concepts	Implementing scaling and encoding
4	Visualize dataset features using Python libraries	Introduction to matplotlib and seaborn for data visualization	Plotting graphs and charts from datasets
5	Apply Linear Regression for predictive modeling	Theory of regression and line fitting	Implementation of Linear Regression
6	Implement Logistic Regression for binary classification	Theory of classification and sigmoid function	Implementation of Logistic Regression
7	Classify data using k-Nearest Neighbors	Concept of distance-based classification	k-NN model building and testing
8	Construct a Decision Tree for classification	Decision tree theory and splitting criteria	Building and visualizing a Decision

			Tree
9	Apply Naive Bayes for classification	Bayes theorem and probabilistic classification	Implementing Naive Bayes model
10	Implement k-Means clustering	Concept of clustering and centroid-based algorithms	Running k-Means on sample data
11	Evaluate models using cross-validation	Importance of evaluation metrics and validation	Train-test split and K-Fold validation
12	Install and verify deep learning frameworks	Overview of TensorFlow and Keras	Installing and testing DL setup
13	Perform binary classification using ANN	Concepts of binary classification in ANN	Building binary classification model
14	Build an Artificial Neural Network for multi-class classification	ANN architecture and activation functions	Creating ANN for classification
15	Apply dropout and batch normalization	Overfitting and regularization in DL	Adding dropout and normalization layers
16	Use a pre-trained model for image classification	Transfer learning concepts	Loading and using MobileNet for classification
17	Implement BFS and DFS	Uninformed search concepts	BFS and DFS implementation on a simple graph
18	Implement Greedy Best-First Search	Heuristic search concepts	Implementing Greedy search with Manhattan distance
19	Create a simple rule-based system	Knowledge representation basics	Coding a basic rule-based inference system
20	Tokenize and remove stopwords from text	NLP pre-processing concepts	Tokenization and stopword removal using NLTK
21	Perform sentiment analysis using keyword matching	Sentiment analysis concepts	Implementing keyword-based sentiment classifier

22	Detect objects or faces in images	Introduction to computer vision	Face detection using OpenCV and pre-trained cascades

CO-PO/PSO Matrix

CONO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM-508.1	3	2	2	3	1	1	2	3	3	3
CM-508.2	3	3	2	3	1	1	2	3	3	3
CM-508.3	3	3	3	3	1	1	2	3	3	3
CM-508.4	3	3	3	3	1	1	2	3	3	3
CM-508.5	3	3	3	3	1	1	2	3	3	3
Average	3	2.8	2.6	3	1	1	2	3	3	3

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

KEY COMPETENCIES:

S.No	Name of the Experiment	Objectives	Key Competencies
1	Install Python, Jupyter Notebook, and essential AI/ML libraries	Set up programming environment for AI & ML	1. Download Python and Jupyter 2. Install numpy, pandas, matplotlib, scikit-learn, tensorflow, keras, opencv 3. Verify installation
2	Explore datasets using loading and inspection techniques	Load datasets and explore structure	1. Use Pandas to load datasets 2. Inspect data using head(), tail(), describe(), info()
3	Perform feature scaling and encoding	Prepare data for ML algorithms	1. Apply normalization/standardization 2. Perform one-hot encoding
4	Visualize dataset features using Python libraries	Represent data visually	1. Use matplotlib/seaborn to plot bar, line, scatter charts 2. Label and style plots
5	Apply Linear Regression for predictive modeling	Predict continuous values	1. Fit Linear Regression model 2. Predict output values 3. Plot regression line
6	Implement Logistic Regression for binary classification	Predict binary outcomes	1. Fit Logistic Regression model 2. Evaluate predictions
7	Classify data using k-Nearest Neighbors	Classify new data based on nearest neighbors	1. Train k-NN model 2. Predict class labels
8	Construct a Decision Tree for classification	Visual decision rules	1. Train Decision Tree 2. Visualize decision paths
9	Apply Naive Bayes for classification	Classify based on probabilities	1. Train Naive Bayes model

			2. Measure accuracy
10	Implement k-Means clustering	Group unlabeled data	1. Apply k-Means algorithm 2. Visualize clusters
11	Evaluate models using cross-validation	Assess generalization ability	1. Use train-test split 2. Apply K-Fold validation
12	Install and verify deep learning frameworks	Prepare deep learning environment	1. Install TensorFlow/Keras 2. Run test script
13	Perform binary classification using ANN	Apply ANN to two-class problems	1. Build binary ANN 2. Evaluate model
14	Build an Artificial Neural Network for multi-class classification	Implement basic ANN	1. Define ANN layers 2. Compile and train ANN
15	Apply dropout and batch normalization	Prevent overfitting in ANN	1. Add dropout layers 2. Add batch normalization
16	Use a pre-trained model for image classification	Apply transfer learning	1. Load pre-trained model 2. Classify sample images
17	Implement BFS and DFS	Explore graphs systematically	1. Implement BFS 2. Implement DFS
18	Implement Greedy Best-First Search	Use heuristic-based search	1. Define heuristic 2. Implement search
19	Create a simple rule-based system	Implement if-else based reasoning	1. Define rules 2. Implement inference
20	Tokenize and remove stopwords from text	Preprocess text for NLP	1. Tokenize text 2. Remove stopwords
21	Perform sentiment analysis using keyword matching	Classify text sentiment	1. Define positive/negative word lists 2. Analyze sentiment
22	Detect objects or faces in images	Apply basic computer vision	1. Load Haar cascade 2. Detect and display bounding

			boxes

UNIT TEST SYLLABUS

UNIT TEST-I	Learning Outcomes: 1 to 11
UNIT TEST-II	Learning Outcomes: 12 to 22

26CM509P- PROJECT WORK

Course Code	Course Title	No. of Periods / Week	Total No. of Periods	Marks for FA	Marks for SA
26CM509P	PROJECT WORK	6	90	40	60

COURSE OBJECTIVES

Upon completion of this course, students will be able to:

- Inculcate team spirit.
- Apply various software life cycle models.
- Successfully design, develop, and deploy a project.

COURSE OUTCOMES

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

CO No.	Syllabus Code	Course Outcome
CO1	CM-509.1	Identify hardware components and software requirements.
CO2	CM-509.2	Prepare an SRS document based on gathered and analyzed requirements.
CO3	CM-509.3	Design the plan document based on the SRS.
CO4	CM-509.4	Code and test the software based on the design document.
CO5	CM-509.5	Practice software maintenance skills, ensuring quality and reliability.
CO6	CM-509.6	Calculate software metrics like cost, LOC, scheduling, manpower, and other resources.

LEARNING OUTCOMES (Project Execution Steps)

Students are expected to follow a systematic approach to their project, encompassing the following key stages:

1. Conduct a need survey to identify a problem.
2. Gather relevant data and select an appropriate software life cycle model.
3. Estimate project costs, required technology, and resources.
4. Develop a detailed project plan and schedule.
5. Prepare a comprehensive Software Requirement Specification (SRS) document.
6. Design the system architecture and elements using standard models like UML.
7. Develop the software modules as per the design.
8. Thoroughly test, debug, and validate the project to ensure it meets requirements.
9. Record results and prepare the final project report and, if applicable, a user manual.

CO-PO/PSO Matrix:

CO NO.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CM-509.1	3	2	1	3	1			2	3	
CM-509.2	3	2	3	1	2	1		2	3	2
CM-509.3	3	2	3	1	2	1		2	3	2
CM-509.4	3	2	3	3	3	1		2	3	2
CM-509.5	3		2	2	3	3	3	2	3	2
CM-509.6	3	2		2	1	3	3	2	3	2
Average	3	2	2.4	2	2.2	1.8	3	2	3	2

3=strongly mapped, 2=moderately mapped, 1=slightly mapped

EVALUATION SCHEME

The project will be evaluated based on the successful completion and demonstration of various tasks throughout its lifecycle.

S.No	Task	Internal Marks (40)	External Marks (60)
1.	Feasibility Study of the Problem	4	6
2.	Requirement Analysis & SRS Document Preparation	4	8
3.	Designing the Problem	6	10
4.	Implementation	8	10
5.	Testing and Verification	10	16
6.	Project Report Preparation and Presentation	8	10
	Total	**40**	**60**

Internal Assessment Scheme (40 Marks)

The 40 internal marks are split between two reviews to assess the project's progress at critical stages. The first review focuses on planning and design, while the second evaluates the implementation, testing, and final delivery.

First Project Review (14 Marks)

This review assesses the initial planning, analysis, and design phases of the project.

Task	Mapping to Project Stage	Max. Marks
Feasibility Study	Assesses the viability and initial plan for the project.	4
Requirement Analysis & SRS	Defines the project scope, functionalities, and deliverables in the SRS document.	4
Designing the Problem	Evaluates the system architecture, database design, and module planning (e.g., using UML diagrams).	6
Total for Review 1		14

Second Project Review (26 Marks)

This review focuses on the development, execution, and final presentation of the completed project.

Task	Mapping to Project Stage	Max. Marks
Implementation	Assesses the quality of the code, adherence to design, and functional modules.	8
Testing and Verification	Evaluates the thoroughness of testing, bug fixing, and validation against the SRS.	10
Project Report & Presentation	Assesses the final documentation, project demonstration, and presentation skills.	8
Total for Review 2		26

Summary of Internal Marks Distribution

Review	Total Marks
First Project Review	14
Second Project Review	26
Grand Total	40

Suggested Project Domains & Ideas (Aligned with C26 Curriculum)

Projects can be chosen from the following domains. These categories are directly linked to the subjects you have studied, allowing you to apply your specialized knowledge.

1. Full-Stack Web and Database Applications

- **Relevant Subjects**: WEB TECHNOLOGIES, DBMS, OOP THROUGH JAVA, PYTHON PROGRAMMING, UI/UX DESIGN.
- **Project Ideas**:
  - **E-commerce Platform**: Develop a complete online store with user accounts, product catalogs, a shopping cart, and an integrated payment gateway.
  - **Online Learning Management System (LMS)**: Create a portal for students and teachers to manage courses, submit assignments, and track progress.
  - **Real-time Polling/Survey Application**: Build a web app where users can create polls and see live results, similar to Mentimeter or Kahoot!
  - **Hostel or Mess Management System**: An application for managing student accommodations, fee payments, and meal preferences.

2. AI/ML and Data Science Projects

- **Relevant Subjects**: FUNDAMENTALS OF AIML, AI USING PYTHON, DATA STRUCTURES THROUGH PYTHON.
- **Project Ideas**:
  - **Student Performance Predictor**: Use college data to build a model that predicts student academic outcomes, helping identify at-risk students early.
  - **College Admissions Chatbot**: Develop an NLP-based chatbot to answer prospective students' questions about courses, fees, and campus life.
  - **Sentiment Analysis of Social Media**: Create a tool to analyze public sentiment on a specific topic (e.g., a new policy or event) by scraping data from social media.
  - **Handwritten Digit or Object Recognizer**: Build and train a neural network to recognize handwritten text or objects in images.

3. Mobile and Android Applications

- **Relevant Subjects**: ANDROID PROGRAMMING, JAVA, MOBILE COMMUNICATIONS, UI/UX DESIGN.
- **Project Ideas**:
  - **Campus Navigator App**: An app that provides indoor maps and navigation for the college campus, helping new students find labs, classrooms, and offices.
  - **Event Notification App**: A mobile application for the college that sends real-time notifications about events, workshops, and placement drives.

    - **Personal Finance Tracker**: Develop an Android app to help users track their income and expenses, with features for budgeting and generating reports.
    - **Fitness and Wellness App**: An app to track daily physical activity, set fitness goals, and provide simple workout routines.

4. Internet of Things (IoT) and Embedded Systems

- **Relevant Subjects**: IoT, COMPUTER ORGANIZATION, DIGITAL ELECTRONICS.
- **Project Ideas**:
    - **Smart Campus System**: Design an IoT network to monitor and control classroom lights and fans based on occupancy, saving energy.
    - **Automated Irrigation System**: Create a system using soil moisture sensors to automatically water plants, with a dashboard to monitor water usage. * **Smart Parking Solution**: Develop a system using sensors to detect empty parking spots and an app to guide drivers to them.
    - **Environment Monitoring Station**: Build a device that measures air quality (PM2.5, CO2) and temperature, and uploads the data to a cloud platform for analysis.

5. Networking and Cyber Security Projects

- **Relevant Subjects**: COMPUTER NETWORKS, CYBER SECURITY, LINUX PRACTICALS, TROUBLESHOOTING OF COMPUTER NETWORKS.
- **Project Ideas**:
    - **Phishing Website Detector**: Create a browser extension or tool that analyzes URLs and web page content to detect and warn users about potential phishing scams.
    - **Secure File Transfer Utility**: Develop a client-server application for transferring files securely over a network using encryption algorithms.
    - **Network Intrusion Detection System (NIDS)**: Design a basic NIDS that monitors network traffic for suspicious patterns or signatures of common attacks.
    - **Folder/File Encryption Tool**: A desktop utility that allows users to encrypt and password-protect their sensitive files and folders.

CURRICULUM-2026

DIPLOMA IN COMPUTER ENGINEERING

ENGINEERING SCHEME OF INSTRUCTIONS AND EXAMINATION

(VI Semester)

CM-601 Industrial Training

Course Code	Course title	Assessment	Periods/ hours	Duration	Marks for FA	Marks for SA
26CM601I	INDUSTRIALTRAINING (Online Certificate courses /Industry)	I	315	90 working days	240	60
		II	315			
TOTAL			630			

LEARNING OUTCOMES (In Industry): The student shall be able to display the following skill sets

1. Apply knowledge and skill already learnt in the institution.
2. Acquire the required skills of analysis, design and development, testing, verification and Validation.
3. Acquire skills of deployment and distribution of the product.
4. Involve in product design, development, and quality testing and maintenance production by Exhibiting the strength, teamwork spirit and self-confidence
5. Prepare product documents like user manual ,installation guide and operational manuals.
6. Perform the activities of deploying product at customer site and training the enduser.
7. Maintaining the system at user site (Post product services)

SNo	UnitTitle	Duration	COsMapped
1	Application of Knowledge acquired.	1month	CO1
2	Skill Acquirement.	2months	CO2
3	Participate in product development.	2months	CO3
4	Preform onsite service.	1month	CO4
	Total	6months	

Course Outcomes

At the end of course student able to:		
CO1	**CM-601.1**	Apply knowledge and skill already learnt in the institution.
CO2	**CM-601.2**	Acquire the required skills of analysis, design and development, testing, verification and validation, deployment and distribution of the product.
CO3	**CM-601.3**	Involve in product design, development, quality testing and maintenance production by exhibiting the strength, teamwork spirit and self-confidence
CO4	**CM-601.4**	Prepare product document, gain the skills in deploying product at customer site , training the end user, maintaining the system.

LEARNINGOUTCOMES–SCHEMOFEVALUATION (Two Online Certificate courses):

TRAINING MODULE NO.	TOPIC	LEARNING OUTCOMES (In-housetraining)	MARKS
First3 Months/12 weeks	1) Registration at Nptel/Swayam/Moocs/ course era/lectera/caltech/NASSCOM/AP SKILL DEVELOPMENT/oxford/ hackerrank/udemy …etc.,	i) Learning ii) Mini Application development iii) Report preparation iv) 1st Assessment	120
Next3 Months/12 Weeks	1) Registration at Nptel/Swayam/Moocs/courseera/lectera/caltech/NASSCOM/AP SKILL DEVELOPMENT/oxford/ hackerrank/udemy…etc.,	i) Learning ii) Mini Application development iii) Report preparation iv) 2st Assessment	120
External Evaluation	Seminar on two reports/viva	Evaluation by GUIDE/Co-Examiner, HOD and External Examiner	60
		TOTAL	300

Scheme of evaluation (Training at Industry)

Sl. No.	Subject	Duration	Scheme of evaluation		
			Item	Nature	Max. Marks
1	Industrial Training	6months	1.First Assessment at Industry(After 12Weeks)	Assessment of learning Out comes by Both the Faculty and training Mentor of the industry	120
			2.Second Assessment at the Industry (After 20 weeks))	Assessment of learning outcomes by both the faculty and training mentor of the industry	120
			Final Summative Assessment at institution level	Training Report	20
				Demonstration of any one of The skills listed In learning outcomes	30
				Viva Voce	10
TOTAL MARKS					300

The industrial training shall carry 300 marks and pass marks are50%. A candidate failing to secure the minimum marks should complete it at his own expenses.

During Industrial training the candidate shall put in a minimum of 90% attendance.

Weightage of marks for Assessment of Learning Outcomes during first and second assessment (at industry)

Sl. No	Learning Outcome	Max Marks Allotted For first assessment	Max Marks Allotted For second assessment
1	Apply knowledge and skill already learnt in the institution.	50	10
2	Acquire the required skills of analysis, design and development, testing, verification and validation , deployment and distribution of the product.	70	30
3	Involve in product design, development, quality testing and maintenance production by exhibiting the strength, teamwork spirit and self-confidence	-	40
4	Prepare product document, gain the skills in deploying product at customer site, training the end user, maintaining the system.	-	40
	Total	120	120

During assessment he performance of the students shall be assessed in those skills in which the student has been trained and be awarded the marks asper the weightage assigned as above. In case the student has undergone training in a few skillsets then the total marks obtained shall be raised to 120marks for the given assessment i.e. either assessment 1 or 2. However the performance of the student shall be assessed at the most skill sets listed above but not less than three skillsets.

Illustration for First assessment:

If the student has undergone training in only in 2skillsets (namely 1 for 50 marks, and 2 for 40 marks) out of 3(namely 1 for 50marks,2 for 40marks and 3 for 30marks) in First assessment and marks awarded during assessment is 60 out of 90marks, then the marks of 60 shall be enhanced to120 proportionately as (60/90) *120=80.

Illustration for second assessment:

If the student has undergone training in only in 5 skillsets (namely 1 for 10 marks,2 for 20 marks,3 for10 marks,4 for 25 marks,5 For 15 marks) out of 7(namely 1 for 10 marks,2 for 20 marks,3 For 10 marks,4 for 25 marks,5 For 15 marks,6 for 25 marks and 7 for15 marks) in second assessment and marks awarded during assessment is 65 out of 80 marks, then the marks of 65 shall be enhanced to 120 proportionately as (65/80) *120=97.5=rounded to 98.

GUIDELINES FOR INDUSTRIALTRAINING OF DIPLOMAIN COMMUNICATION ANDNETWORKING ENGINEERING PROGRAMME

1. Duration of the training:6months.

2. Eligibility: The as per SBTET norms

3. Training Area: Students can be trained in either in In−house/Industry in the areas of

4. Application Software Development / system software Development / firmware development/ Mobile application development/ Database applications / Web development/ IoT application development / smart technologies / Hardware interfacing/ Networking.

5. The candidate shall put a minimum of 90% attendance during Industrial Training.

6. If the student fails to secure 90% attendance during industrial training, the student shall reappear for 6 months industrial training.

7. Formative assessment at industry level shall be carried out by the Mentor from of the industry, where the student is undergoing training and the faculty in charge (Guide)from the concerned section in the institution.

8. The Industrial training shall carry 300 marks and pass marks is 50% in assessments at industry (first and second assessment) and final summative assessment at institution level put together i.e. 150 marks out of 300 marks.

9. If the student fails to secure 50% marks in final summative assessment at institution level, the student should reappear for final summative assessment in the subsequent board examination.

10. Final summative assessment at institution level is done by a committee including1. Head of the section (of concerned discipline ONLY), 2. External examiner from an industry and 3. Faculty member who assessed the student during Industrial Training as members.

Guidelines and responsibilities of the faculty members who are assessing the students'

performance during industrial training

DEPARTMENT OF TECHNICAL EDUCATION
NAME OF THE INSTITUTION
INDUSTRIALTRAININGFIRST ASSESSMENT

PIN:

NAME OF THE STUDENT:

Name of the Industry:

Skill Set Sl. No	**SKILL SET**	**Max Marks Allotted For each parameter**	**Marks obtained**
1	Apply knowledge and skill already learnt in the institution.	50	
2	Acquire the required skills of analysis, design and development ,testing, verification and validation.	40	
3	Acquire the required skills of deployment and distribution of the product.	30	
	Total	120	

(Marks in words:)

Signature of the Training In−charge(Mentor)

Name:

Designation:

Signature of the visiting staff(Guide)

Name:

Designation:

DEPARTMENTOF TECHNICAL EDUCATION NAME OF THE INSTITUTION

INDUSTRIAL TRAINING SECOND ASSESSMENT

PIN:

NAME OF THE STUDENT:

Name of the Industry:

Skill Set Sl. No	**SKILL SET**	**Max Marks Allotted For Each parameter**	**Marks obtained**
1	Apply knowledge and skill already learnt in the institution.	10	
2	Acquire the required skills of analysis, design and development ,testing, verification and validation.	20	
3	Acquire the required skills of deployment and distribution of the product.	10	
4	Involve in product design, development, quality testing and maintenance production by exhibiting the strength, teamwork spirit and self– confidence	25	
5	Prepare product documents like user manual and installation guide and operational manuals.	15	
6	Perform the activities of deploying product at customer site and training the end user.	25	
7	Maintaining the system at user site(Post product services)	15	
		120	

(Marks in words:)

Signature of the Training In–charge (Mentor) Signature of the visiting staff(Guide) Name: Name:

Designation: Designation: