DAYANANDA SAGAR UNIVERSITY

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bangalore South Dt., Bengaluru – 562 112

Department of

Computer Science & Technology

B.Tech. PROGRAMME– 2025 BATCH

SCHEME AND SYLLABUS

(1th to 8th semester)

Academic Year 2025-26

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

PART-B

SCHEME AND SYLLABUS FOR B.TECH. CST (FIRST TO FOURTH YEARS) PROGRAMME

Definitions / Descriptions

Definition of Credit:	
1 Hour Lecture (L) Per Week	01 Credit
1 Hour Tutorial (T) Per Week	0.5 Credit
1 Hour Practical (P) Per Week	0.5 Credit
1 Hour Project (J) Per Week	0.5 Credit

Course code and Definition:	
BSC	Basic Science Courses
ESC	Engineering Science Courses
HSMC	Humanities and Social Sciences including Management Courses
IPCC	Integrated Professional Core Course
PCC	Professional Core Courses
PEC	Professional Elective Courses
OEC	Open Elective Courses
SEC	Skill Enhancement Courses
UHV	Universal Human Value Course
PROJ	Project Work
INT	Internship

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

SCHEME 2025 – 2026 Batch

III SEMESTER													
S.N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	BSC	25CT2301	Transforms and Numerical Techniques	MAT	3	0	0	0	03	60	40	100	**3**
2	IPCC	25CT2302	Data Structures	CSE	3	0	2	0	03	60	40	100	**4**
3	IPCC	25CT2303	Digital Logic Design	ECE	3	0	2	0	03	60	40	100	**4**
4	PCC	25CT2304	Discrete Mathematics and Graph Theory	CSE	3	0	0	0	03	60	40	100	**3**
5	PCC	25CT2305	Full Stack Development	CST	3	0	0	2	03	60	40	100	**4**
6	SEC	25CT23XX	Skill Enhancement Course – I	CST	1	0	0	2	01	100	--	100	**2**
7	SEC	25CT2309	Product Marketing	CST	1	0	0	2	01	100	--	100	**2**
8	SEC	25CT2310	Cognitive And Technical Skills – III	Any Dept.	0	0	0	0	00	--	--	--	**P/F**
			Total		**17**	**00**	**04**	**06**	**17**	**500**	**200**	**700**	**22**

Skill Enhancement Course – I		
S.N	**Course Code**	**Course Name**
1	25CT2306	Design Thinking & Ideation
2	25CT2307	Innovative Businesses & Breakthrough Technologies
3	25CT2308	R for Data Analysis and Application Development

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

IV SEMESTER													
S.N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week			Examination					Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	BSC	25CT2401	Probability & Statistics	MAT	3	0	0	0	03	60	40	100	**3**
2	IPCC	25CT2402	Design and Analysis of Algorithms	CSE	3	0	2	0	03	60	40	100	**4**
3	IPCC	25CT2403	Database Management System	CSE	3	0	2	0	03	60	40	100	**4**
4	PCC	25CT2404	Introduction to Artificial Intelligence	CSE	3	0	0	0	03	60	40	100	3
5	PCC	25CT2405	Computer Organization and Architecture	CSE	3	0	0	0	03	60	40	100	**3**
6	SEC	25CT24XX	Skill Enhancement Course – II	CST	1	0	0	2	01	100	--	100	**2**
7	SEC	25CT2410	DevOps Fundamentals	CST	1	0	0	2	01	100	--	100	**2**
8	SEC	25CT2411	Cognitive And Technical Skills - IV	CST	0	0	0	0	--	--	--	--	**P/ F**
Total					**17**	**00**	**04**	**04**	**16**	**500**	**200**	**700**	**21**

Skill Enhancement Course – II		
S.N	**Course Code**	**Course Name**
1	25CT2406	Small E-Business Launch
2	25CT2407	Idea Generation and Validation
3	25CT2408	Technical writing
4	25CT2409	Introduction to IPR & Registration

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

V SEMESTER													
S.N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	PCC	25CT3501	Theory of Computation	CSE	3	1	0	0	03	60	40	100	**4**
2	PCC	25CT3502	Secure Software Engineering & Project Management	CSE	3	0	0	0	03	60	40	100	**3**
3	IPCC	25CT3503	Machine Learning	CSE	3	0	2	0	03	60	40	100	**4**
4	IPCC	25CT3504	Product Design & Development	CST	2	0	0	2	03	60	40	100	**3**
5	IPCC	25CT3505	Operating System	CSE	3	0	2	0	03	60	40	100	**4**
6	PEC	25CT35XX	Professional Elective Course – I	CST	3	0	0	0	03	60	40	100	**3**
7	SEC	25CT35XX	Skill Enhancement Course – III	CST	1	0	0	2	01	100	--	100	**2**
8	SEC	25CT3512	Cognitive And Technical Skills - V	CST	0	0	0	0	00	--	--	--	**P/F**
			Total		**18**	**01**	**04**	**04**	**19**	**460**	**240**	**700**	**23**

NOTE: For Professional Elective Courses (PEC), the LTPJ division is given on the next page according to the type of course.

Professional Elective Course – I / MOOC/ Certification						
S.N	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	25CT3506	Interactive Art and Creative Coding	2	0	0	2
2	25CT3507	Quantum Computing Fundamentals	3	0	0	0
3	25CT3508	IoT Fundamentals: Architecture to Analytics	2	0	0	2
4	25CT3509	NoSQL Database Fundamentals	2	0	0	2

Skill Enhancement Course – III		
S.N	**Course Code**	**Course Name**
1	25CT3510	DevSecOps Fundamentals and Best Practices
2	25CT3511	AWS Web Services

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

VI SEMESTER													
S.N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	HSMC	25CT3601	Supply Chain Management and Operations	CST	2	0	0	0	02	60	40	100	**2**
2	IPCC	25CT3602	Introduction to Computer Networks	CST	3	0	2	0	03	60	40	100	**4**
3	IPCC	25CT3603	Lean Startup Methodology	CST	3	0	0	2	03	60	40	100	**4**
4	OEC	25OEXXXX	Open Elective – I	All.Depts	3	0	0	0	03	60	40	100	**3**
5	PEC	25CT36XX	Professional Elective Course – II	CST	3	0	0	0	03	60	40	100	**3**
6	PEC	25CT36XX	Professional Elective Course – III	CST	3	0	0	0	03	60	40	100	**3**
7	PROJ	25CT3612	Minor Project	CST	0	0	0	4	01	100	--	100	**2**
8	SEC	25CT3613	Cognitive And Technical Skills- VI	CST	0	0	0	0	00	100	--	100	**P/F**
			Total		**17**	**00**	**02**	**06**	**18**	**560**	**240**	**800**	**21**

NOTE1: For Professional Elective Courses (PEC), the LTPJ division is given on the next page according to the type of course.

NOTE2:

Internship: All the students admitted to III year shall have to undergo mandatory internship of 4 weeks during the vacation of VI and VII semesters and /or VII and VIII semesters. A university examination shall be conducted during VIII semester and the prescribed credit shall be included in VIII semester. Internship shall be considered as a head of passing and shall be considered for the award of degree. Those, who do not take up/complete the internship shall be declared fail and shall have to complete during subsequent University examination after satisfying the internship requirements

Open Electives-I		
S.N	**Course Code**	**Course Name**
1	XXXXXX	Small E-Business Launch
2	XXXXXX	Product Engineering & Entrepreneurship

Professional Elective Course – II / MOOC/ Certification						
S.N	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	25CT3604	Gaming Design Fundamentals	2	0	0	2
2	25CT3605	Fundamentals of Quantum Cryptography/ Quantum Algorithms and Cryptography	3	0	0	0
3	25CT3606	Edge Computing with IoT	2	0	0	2
4	25CT3607	Data Engineering / Data Science for Engineers	2	0	0	2

Professional Elective Course – III / MOOC/ Certification						
S.N	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	25CT3608	Game Development using Unity/ Game Development Using Godot	2	0	0	2
2	25CT3609	Qiskit and Quantum Circuits: A Hands-On Approach	2	0	0	2
3	25CT3610	Building Intelligent IoT Systems/ Foundations of Cloud IoT Edge ML	2	0	0	2
4	25CT3611	Effective Data Transformation with DBT	2	0	0	2

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

VII SEMESTER													
S.N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	IPCC	25CT4701	Product Analytics	CST	3	0	0	2	03	60	40	100	**4**
3	OEC	25OEXXXX	Open Elective – II	All. Dept.	3	0	0	0	03	60	40	100	**3**
4	PEC	25CT47XX	Professional Elective Course – IV	CST	3	0	0	0	03	60	40	100	**3**
5	PEC	25CT47XX	Professional Elective Course – V	CST	3	0	0	0	03	60	40	100	**3**
6	PROJ	25CT4710	Capstone Project-Phase I	CST	0	0	0	10	03	100	--	100	**5**
			Total		**11**	**00**	**00**	**12**	**15**	**340**	**160**	**500**	**18**

NOTE: For Professional Elective Courses (PEC), the LTPJ division is given on the next page according to the type of course.

Open Electives-II			
S.N	**Course Code**	**Course Name**	
1	XXXXX	Lean Startup Methodology	
2	XXXXX	Product Analytics	

Professional Elective Course – IV / MOOC/ Certification						
S.N	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	25CT4702	AR/VR & Game-Theoretic/ Algorithmic Game Theory	2	0	0	2
2	25CT4703	Quantum Artificial Intelligence	2	0	0	2
3	25CT4704	Cyber-physical systems /Foundations of Cyber-physical Systems	2	0	0	2
4	25CT4705	Introduction to Data Visualization	2	0	0	2

Professional Elective Course – V / MOOC/ Certification						
S.N	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	25CT4706	Unreal Engine Game Development for Beginners	2	0	0	2
2	25CT4707	Quantum Computing with PennyLane/ Quantum Machine learning with TensorFlow Quantum	2	0	0	2
3	25CT4708	Real-Time Cyber-Physical Systems Design using MATLAB	2	0	0	2
4	25CT4709	Tableau for Data Analysts/ Power BI for Data Analysts	2	0	0	2

VIII SEMESTER													
S.N	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	PROJ	25CT4801	Capstone Project-Phase II	CST	0	0	0	24	03	60	40	100	**12**
2	INT	25CT4802	Internship	--	0	0	6	0	03	100	--	100	**03**
			Total		**00**	**00**	**06**	**24**	**06**	**160**	**40**	**200**	**15**

NOTE 1: Internship

Completed during the intervening vacations of VI and VII semesters and /or VII and VIII semesters or VIII semesters

NOTE: Total Credits (I-Sem to VIII Sem) = 160 credits.

I -20 V -23
II -20 VI -21
III -22 VII - 18
IV - 21 VIII – 15

Total=160

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

Elective courses domain-wise

S. N	Domain-wise	Domain Clusters	PROFESSIONAL ELECTIVE COURSES				
			PEC-I	PEC-II	PEC-III	PEC-IV	PEC-V
			5th Semester	6th Semester		7th Semester	
			Course Name	Course Name	Course Name	Course Name	Course Name
1	Domain-1	GAMING AND ANIMATIONS/ AR&VR	Interactive Art and Creative Coding	Gaming Design Fundamentals	Game Development using Unity/ Game Development Using Godot	AR/VR & Game-Theoretic/ Algorithmic Game Theory	Unreal Engine Game Development for Beginners
2	Domain-2	QUANTUM INFORMATION SYSTEM	Quantum Computing Fundamentals	Fundamentals of Quantum Cryptography/ Quantum Algorithms and Cryptography	Qiskit and Quantum Circuits: A Hands-On Approach	Quantum Artificial Intelligence	Quantum Computing with PennyLane/ Quantum Machine learning with TensorFlow Quantum
3	Domain-3	INTERNET OF THINGS TOOLS & TECHNOLOGIES	IoT Fundamentals : Architecture to Analytics	Edge Computing with IoT	Building Intelligent IoT Systems/ Foundations of Cloud IoT Edge ML	Cyber-physical systems /Foundations of Cyber-physical systems	Real-Time Cyber-Physical Systems Design using MATLAB
4	Domain-4	DATA ANALYTICS	NoSQL Database Fundamentals	Data Engineering / Data Science for Engineers	Effective Data Transformation with DBT	Introduction to Data Visualization	Tableau for Data Analysts/ Power BI for Data Analysts

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

TRANSFORMS AND NUMERICAL TECHNIQUES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Subject Code	:	25CT2301	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P-J	:	3–0–0-0			

Course Learning Objectives:

This Course will enable students to:

1. **Apply** their knowledge of Laplace transforms and inverse Laplace transforms to proficiently solve linear ordinary differential equations with constant coefficients, facilitating the analysis and modelling of complex systems.
2. **Analyze** periodic functions using Fourier series, assessing the convergence properties and precision of the series expansion, thereby enhancing their ability to understand and manipulate periodic phenomena.
3. **Utilize** complex exponential form, Fourier transforms of basic functions, and Fouriersine and cosine transforms to solve problems involving Fourier integrals, developingproficiency in applying these techniques to various mathematical scenarios.
4. **Employ** numerical methods, including Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods, to solve differential equations and effectively analyze dynamic systems, enabling them to model real-world phenomenaand make accurate predictions**.**
5. **Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to effectively solve different types of partial differential equations (PDEs) such as elliptic, parabolic, and hyperbolic equations, enhancing their problem-solving skills in the context of differential equations and their applications

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I: Laplace Transform and Inverse Laplace Transform	**09 Hours**

Laplace Transforms of Elementary functions (without proof),
(Text Book-1: Chapter 6: 203 to 207).
Laplace Transforms of $e^{at}f(t)$, $t^n f(t)$ and $\frac{f(t)}{t}$, Periodic functions, Unit step function and impulse functions
(Text Book-1: Chapter 6:208-230).
Inverse Laplace Transforms- By the method of Partial Fractions, Logarithmic and Trigonometric functions, Convolution Theorem, Inverse Laplace transform using Convolution Theorem ***(Text Book-1: Chapter 6: 238)***.
Solution to Differential Equations by Laplace Transform.
(Text Book-1: Chapter 238-242).

UNIT – II: Fourier Series	**09 Hours**

Periodic Functions, Trigonometric Series
(Text Book-1: Chapter 11: 495).
Fourier series Standard function, Functions of any Period 2L, Even and Odd functions, Half-range Expansions.
(Text Book-1: Chapter 11: 483-492)
Practical Harmonic analysis (calculate average power and RMS values of periodicwaveforms)

UNIT – III: Fourier Transform	**06 Hours**
Calculation of Fourier integrals using complex exponential form ***(Text Book-1: Chapter 11: 510).*** Fourier transform of basic functions ***(Text Book-1: Chapter 11: 510-516).*** Fourier sine and cosine transforms. ***(Text Book-1: Chapter 11: 518-522).***	
UNIT – IV: Numerical Methods for Solving Ordinary DifferentialEquations	**07 Hours**
Euler's Method-Basic principles of Euler's method for solving first-order ODEs ***(Text Book-1: Chapter 1:10-12).*** Runge-Kutta 4th order ***(Text Book-1: Chapter 21:904).*** Multistep Methods-Explanation of multistep methods (Adams-Bashforth, Adams–Moulton Methods) ***(Text Book-1: Chapter 21:911-913).*** Second-Order ODE. Mass–Spring System (Euler Method, Runge–Kutta Methods) ***(Text Book-1: Chapter 21:916-918).***	
UNIT – V: Numerical Methods for Partial Differential Equations	**08 Hours**
Classification of PDEs (elliptic, parabolic, hyperbolic), ***(Text Book-1: Chapter 21:922-923).*** Finite Difference Methods (Laplace and Poisson Equations), Derivation of finite difference approximations ***(Text Book-1: Chapter 21:923-927).*** Crank–Nicolson Method ***(Text Book-1: Chapter 21:938-941).*** Method for Hyperbolic PDEs ***(Text Book-1: Chapter 21:943-945).***	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Apply** Laplace transforms and inverse Laplace transforms tosolve linear ordinary differential equations with constant coefficients, demonstrating proficiency in system analysis andmodelling.	L3
2	**Analyze** periodic functions using Fourier series and evaluate the convergence properties and precision of the seriesexpansion.	L2 & L3
3	**Solve** problems involving Fourier integrals by applying complex exponential form, Fourier transforms of basicfunctions, and Fourier sine and cosine transforms.	L3

4	**Utilize** numerical methods such as Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods to solve differential equations and analyze dynamic systems	L2 & L3
5	**Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to solve various types of partial differential equations (PDEs)such as elliptic, parabolic, and hyperbolic equations.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	2	1					1				
CO2	3	2	2						1				
CO3	3	2	2	1					1				
CO4	3	2	2	1					1				
CO5	3	2	2	1					1				

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 2015, 10th Edition, Wiley India.

REFERENCE BOOKS:

1. Higher Engineering Mathematics, B.S. Grewal, 2015, 43rd Edition, Khanna Publishers.
2. Higher Engineering Mathematics, John Bird, 2017, 6 th Edition, Elsevier Limited.

E-Resources:

1. https://nptel.ac.in/courses/111106139
2. https://nptel.ac.in/courses/111101164
3. https://nptel.ac.in/courses/111105038

****

DATA STRUCTURES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	:	25CT2302	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39(Th)+26(P) Hours
L–T–P-J	:	3–0–2-0			

Prerequisites:

Proficiency in a C programming language.

Course Objectives:

This Course will enable students to:

1. **Understand** the basic approaches for analyzing and designing data structures.
2. **Introduce** dynamic memory allocation and C language concepts required for building data structures.
3. **Develop** essential skills to construct data structures to store and retrieve data quickly and **efficiently**.
4. **Utilize** different data structures that support different sets of operations which are suitable for various applications.
5. **Explore & implement** how to insert, delete, search, and modify data in any data structure-Stack, Queues, Lists, Trees.
6. **Develop** applications using the available data structure as part of the course for mini project.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	08 Hours
IINTRODUCTION: Introduction to Data Structure, Classification, C Structure and Union, C Pointers, Array Definition, Representation, Operations (Insertion, Deletion, Search and Traversal), Two/Multidimensional Arrays, Sparse matrix. TB1: 1.1, 2.2, 2.5 ; TB2: 1.1, 1.2,1.3.1-1.3.4; RB1: 5.1 – 5.12, 6.4	
UNIT – II	**08 Hours**
INTRODUCTION TO ADT: **Stack:** Definition, Array Representation of Stack, Operations on Stacks. **Applications of Stack:** Expression evaluation, Conversion of Infix to Postfix, Infix to Prefix Recursion, Tower of Hanoi **Queue:** Definition, Representation of Queues, Operations of Queues, Circular Queue. **Applications of Queue:** Job Scheduling, A Maze Problem TB1: 3.1,3.2, 3.3,3.4,3.5 ; TB2: 2.1, 2.2, 2.3, 3.2, 3.3	
UNIT – III	**08 Hours**
DYNAMIC DATA STRUCTURES: **Linked List:** Types, Representation of Linked Lists in Memory. Traversing, Searching, Insertion & Deletion from Linked List. Circular List, Doubly Linked List, Operations on Doubly Linked List (Insertion, Deletion, Traversal). **Applications:** Stack & Queue Implementation using Linked Lists. **Case Study**: Josephus problem. TB2: 4.2,4.3,4.5	
UNIT – IV	**08 Hours**
TREES: Basic Terminology, Binary Trees and their representation, Complete Binary Trees, Binary Search Trees, Threaded Binary Trees, Operations on Binary Trees (Insertion, Deletion, Search & Traversal). **Applications:** Expression Evaluation **Case Study:** Game Tree TB2: 5.5.3,5.5.4,5.6	
UNIT – V	**07 Hours**
Efficient Binary Search Trees: Optimal Binary Search Trees, AVL Trees, Red Black Trees, Splay Trees. **Case Study**: B Trees TB1: 10.1,10.2,10.3,10.4, 11.2	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Apply the concepts of pointers, arrays, structures, and unions to address real-world problems and implement the concept in C programming language.	L3
2	Utilize stacks and queue data structures to solve problems such as infix to postfix, infix to prefix conversions, the Towers of Hanoi puzzle, job scheduling and maze navigation.	L3
3	Implement and manipulate singly linked lists, doubly linked lists, and circular linked lists, executing operations such as insertion, deletion, and traversal.	L3
4	Understand the concepts of binary trees, binary search trees, and threaded binary trees, and their associated operations.	L2
5	Understand advanced binary tree structures includes optimal binary search trees, AVL trees, Red-Black trees, and Splay trees.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	1	2	3	-	3	-	-	-	-	-	-		2	2
CO2	1	2	3	-	3	-	-	-	-	-	-		2	2
CO3	1	2	3	-	3	-	-	-	-	-	-		2	2
CO4	-	1	2	-	3	-	-	-	-	-	-		2	2
CO5	-	1	2	-	-	-	-	-	-	-	-		2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS (TB):

1. Ellis Horowitz, Susan Anderson-Freed, and Sartaj Sahni, "Fundamentals of Data structures in C", 2nd Edition, Orient Longman, 2008.
2. A.M. Tannenbaum, Y Langsam, M J Augentien "Data Structures using C", 1st Edition, Pearson, 2019.

REFERENCE BOOKS:

1. Brian. W. Kernighan, Dennis. M. Ritchie, "The C Programming Language", 2nd Edition, Prentice-Hall, 1988.
2. Gilbert & Forouzan, "Data Structures: A Pseudo-code approach with C", 2nd Edition, Cengage Learning,2014.
3. Jean-Paul Tremblay & Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, McGraw Hill, 2013.
4. R.L. Kruse, B.P. Learly, C.L. Tondo, "Data Structure and Program design in C", 5th Edition, PHI ,2009.

E-Resources:

1. https://nptel.ac.in/courses/106102064
2. https://www.coursera.org/learn/data-structures?specialization=data-structures-algorithms
3. https://www.udemy.com/topic/data-structures/free/
4. https://www.mygreatlearning.com/academy/learn-for-free/courses/data-structures
5. https://cse01-iiith.vlabs.ac.in/
6. https://kremlin.cc/k&r.pdf

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving using group discussion.
2. Role play E.g., Stack, Queue, etc.,
3. Demonstration of solution to a problem through programming.
4. Flip class activity E.g., arrays, pointers, dynamic memory allocation, etc.,

DATA STRUCTURES LABORATORY

Total Contact Hours: 26

Following are experiments to be carried out using either C programming language.

1. To Implement C programs to perform array operations.
2. To determine the validity of a 9x9 Sudoku board (application of 2-dimensional array).
3. To store, retrieve and update the elements in structures (structures and pointers to structures).
4. To implement stack using linked list.
5. To implement a queue data structure using a singly linked list.
6. To implement a singly linked list and its operations.
7. To implement a doubly linked list and its operations.
8. To create a circular queue using a circular linked list data structure
9. To implement binary tree traversal techniques.

OPEN-ENDED EXPERIMENTS

1. Design a web browser history tracker in C. Implement a stack data structure to keep track of visited URLs. Create functions to push new URLs onto the stack as users visit websites and pop URLs when users navigate backward in their browsing history.
2. Imagine you are responsible for designing a queue-based system to manage the queue of regular customers waiting to purchase cinema tickets at a popular movie theatre. Your system should ensure fair and efficient ticket sales for all customers. When a customer's arrive at the cinema, they join the queue. Each customer is represented by his name, age (for record-keeping), and number of tickets needed. When a customer reaches the front of the queue, they are served by the ticketing agent. Implement a ticket sale process where the agent provides the customer with the requested ticket(s). Initialize the total number of tickets and if the tickets are sold, then the ticketing agent should display a houseful message.

****

DIGITAL LOGIC DESIGN

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Subject Code	:	25CT2303	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39(T)+26(P) Hours
L–T–P–J	:	3–0–2–0			

Course Learning Objectives:

This Course will enable students to:

1. Translate the elements of digital logic functions to digital system abstractions using Verilog.
2. Illustrate simplification of Boolean expressions using Karnaugh
3. Model combinational logic circuits for arithmetic operations and logical operations
4. Analyze and model sequential elements flip-flops, counter, shift registers.
5. Outline the concept of Mealy Model, More Model and apply FSM to solve a given design problem

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION:

Number System- Binary, Hexa, Decimal, Octal and its conversion. Canonical Notation - SOP & POS forms, Minimization of SOP and POS forms.

ARITHMETIC CIRCUITS AND VERILOG MODELLING

Adders: Half adder, full adder, Ripple carry adder, parallel adder /subtractor, fast adders-CLA, comparator- 2 bit. Simplification using K-Maps

Introduction to Verilog, Syntax of Verilog coding, Modelling styles in Verilog, Verilog Operators, Test bench for simulation

Text Book-1: Chapter 1: 1.2 to 1.4, Chapter 2: 2.6

Text Book-2: Chapter 5: 5.2, 5.3.3, 5.4,5.5.2, 5.5.3

Text Book-3: Chapter 1: 1.1, 1.2.2, 1.3.1, 1.3.2, 1.3.3, 1.4.2, 1.5.1.2, 1.5.2.2, 1.5.3.2, 1.5.4.2, 1.6.2	
UNIT – II	**07 Hours**
Combinational Circuit Building Multiplexers 4:1, 8:1, decoders 3:8, 2:4, demultiplexers 1:4, encoders 8:3, 4:2, code converters- B to G and G to B- Simplification using K-Maps **Verilog for combinational circuits**, if else, case-casex, casez, for loop, generate. **Text Book-2: Chapter 6: 6.1, 6.2, 6.3, 6.4, 6.6**	
UNIT – III	**08 Hours**
Sequential Circuits-1 Basic Latch, Gated latches, Flip Flops SR, D, JK, T, master-slave flip-flops JK, Characteristic equations, 0's and 1's Catching Problem, Race round condition, Switch debounce, shift registers- SISO, SIPO, PISO, PIPO, Setup time, Hold time, Propagation Delay **Text Book-2: Chapter 7: 7.1, 7.2,7.3, 7.4,7.5,7.6, 7.8**	
UNIT – IV	**8 Hours**
Sequential Circuits-2 Binary counters – asynchronous and synchronous, mod-n counter, ripple counter- 4 bit. Verilog blocking and non-blocking, Mealy Model, Moore Model, State machine notation, Construction of Finite State Machine. **Text Book-2: Chapter 7: 7.9, 7.11, 7.12.3, 7.12.4, 8.1, 8.2, 8.3, 8.4**	
UNIT – V	**8 Hours**
Introduction to Electronic Design Automation: FPGA Design Flow, ASIC Design flow, architectural design, logic design, simulation, verification and testing, 3000 Series FPGA architecture. **Applications:** Design 4 Bit ALU, 7 Segment display, Vending Machine, 3 Pipeline. **Text Book-4: Chapter 1**	

Laboratory Experiments
Experiments are conducted using Verilog tool /Kits
1. Introduction to Xilinx tool, FPGA flow
2. Adder – HA, FA using data flow and behavior modelling styles

3.	Adder – HA, FA using structural modelling style
4.	Combinational designs – I (blocking and non-blocking/looping examples) **a.** Multiplexer: 4:1, 8:1 MUX. **b.** De Multiplexer: 1:4, 1:8 DEMUX.
5.	Combinational designs – II (different types of case statements) **c.** Encoder with and without Priority: 8:3 and 4:2. **d.** Decoder: 3:8 and 2:4.
6.	Design of 4-bit ALU
7.	Flip Flop: D FF, T FF, JK FF
8.	Design of Mod – n Up/Down Counter with Synchronous reset
9.	Design of Mod – n Up/Down Counter with Asynchronous reset.
10.	Design of Universal shift Register using FSM

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Interpret Boolean Expressions of digital design in simplified form	L2
2	Build the various elements of digital logic system with Verilog	L3
3	Construct Combinational and Sequential logic circuits	L3
4	Apply the hardware model of a digital system at different levels of abstraction in Verilog	L3
5	Apply the functionality of digital design by implementing on FPGA kits	L4

6	Build digital systems using FSM	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3		1										2	
CO2	3	1	1		1								2	
CO3	3	1	3										2	
CO4	3	1	2		1								2	
CO5	3	2	2		1								2	
CO6	3	2	2										2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. M. Morris Mano Michael D. Ciletti , "Digital Design with an Introduction to the Verilog HDL", 6th Edition, Pearson Education, 2014.
2. Stephen Brown, Zvonko Vranesic, "Fundamentals of Digital Logic with Verilog design", McGraw Hill, 2014.
3. Nazein M. Botros, "HDL programming (VHDL and Verilog)", Dreamtech Press, 2006.
4. Douglas J Smith, "HDL Chip Design", Doone publications 1996.

REFERENCE BOOKS:

1. John M Yarbrough, "Digital Logic Applications and Design", Thomson Learning, 2014.
2. Donald D. Givone, "Digital Principles and Design", McGraw Hill, 2015.
3. Samir Palnitkar, "Verilog HDL: A Guide to Digital Design and Synthesis", Pearson Education, 2016.

E-Resources:

1. https://archive.nptel.ac.in/courses/106/105/106105165/

2. https://nptel.ac.in/courses/117105080

Activity Based Learning (Suggested Activities in Class)

1. Design problem solving and Programming using group discussion. E.g., Traffic light controller, Digital Clock, Elevator.
2. Demonstration of solution to a problem through simulation.

****

DISCRETE MATHEMATICS AND GRAPH THEORY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Subject Code	:	25CT2304	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1**. Learn** the set theoretic concept and its application in theory of computation.
2. **Determine** the concepts of mathematical induction, recursive relations and their application.
3. **Illustrate** the association of functions, relations, partial ordered set and lattices with problems related to theoretical computer science and network models.
4. **Discuss** the basics of graph theory and its application in computer networks. Learn the concepts of counting techniques and its application.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. **Lecture method** means it includes not only traditional lecture method, but different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** Adopt the Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, ask at least three Higher order Thinking questions in the class.
6. Adopt **Problem Based Learning**, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every **concept can be applied to the real world** - and when that possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

SET THEORY: Sets and subsets, Operations on Sets: Basic set operations, algebraic properties of sets, The Addition Principle

RELATIONS AND ITS PROPERTIES: Relations and their properties, N-Ary Relations and their applications, Representing relations. ***Textbook – 2: 1.1, 1.2 ; Textbook – 1: 7.1., 7.2, 7.3***	
UNIT – II	**06 Hours**
RELATIONS AND ORDER RELATIONS: Closure of relations, Equivalence Relations, Partial Orderings, Functions, The Growth of Functions. **Self-Study:** Transitive Closure and Warshall's Algorithm. ***Textbook – 1: 7.4., 7.5, 7.6, 3.2***	
UNIT – III	**08 Hours**
MATHEMATICAL INDUCTION AND RECURSION: Mathematical Induction, Recurrence Relations: Rabbits and the Fibonacci Numbers, The Tower of Hanoi, Code word Enumeration, Solving Linear Recurrence Relations **Self-Study:** Basic Connectives and Truth Tables ***Textbook-1: 4.1;6.1, 6.2;1.1***	
UNIT – IV	**09 Hours**
GRAPH THEORY: Graphs and Graph Models. Graph Terminology and Special Types of Graphs: Basic Terminology, Some Special Simple Graphs, Bipartite Graphs, Complete Bipartite Graphs. Representing Graphs and graph isomorphism: Adjacency lists, Adjacency Matrices, Incidence Matrices, Connectivity: Paths, Connectedness in Undirected and Directed Graphs, Vertex and Edge connectivity and their applications. ***Textbook-1: 8.1, 8.2, 8.3, 8.4***	
UNIT – V	**08 Hours**
GRAPHS AND ITS APPLICATIONS: Euler and Hamilton Paths and their applications, Planar Graphs and their Applications, Graph Coloring and its applications. ***Textbook-1: 8.5, 8.7, 8.8***	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Identify** the membership of the Set, Relations and make use of basic Algebraic properties.	L3

2	**Examine** the steps involved in Mathematical Induction and Linear recurrence-related problems.	L4
3	**Construct** different types of graphs based on the properties and the real-time applications of graph theoretical concepts.	L3
4	**Analyze** the methods for optimizing the solution for graph coloring problem, Eulerian and Hamilton circuits/planes.	L4

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	1	2					1	1	1			1	0
CO2	3	3	2					1	1	1			1	0
CO3	3	3	3					1	1	1			1	0
CO4	3	3	3					1	1	1			1	0

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Kenneth H. Rosen, "Discrete Mathematics and its applications", Tata McGraw Hill, 2003.
2. Bernard Kolman, Robert C. Busby, Sharon Ross, "Discrete Mathematical Structures", 3rdEdition, PHI 2001.

REFERENCE BOOKS:

1. Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics", IV Edition, Pearson Education, Asia, 2002.
2. J. P. Tremblay, R. Manohar, "Discrete Mathematical Structures with applications to computer Science", Tata McGraw Hill, 1987.
3. J K Sharma, "Discrete Mathematics", 3rd edition, 2013, Macmillan India Ltd.

E-Resources:

1. Discrete Mathematics with Algorithms by M. O. Albertson, J. P. Hutchinson – J 1988, Wiley.
2. Discrete Mathematics for Computer Science, Gary Haggard, John Schlipf, Sue Whitesides,Thomson Brooks/Cole, 2006.
3. http://ocw.mit.edu/courses/mathematics/
4. http://www.nptelvideos.in/2012/11/discrete-mathematical-structures.html
5. http://cglab.ca/~discmath/notes.html
6. https://www.cs.odu.edu/~toida/nerzic/content/web_course.html

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solution to a problem using graph theory.

****

FULL STACK DEVELOPMENT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	:	25CT2305	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Develop proficiency in using HTML5 to create well-structured, semantically correct web pages. Incorporate multimedia elements, forms, and hyperlinks to enhance interactivity and accessibility.
2. Apply advanced CSS3 techniques to style web pages, ensuring they are visually appealing and responsive. Master the CSS Box Model, selectors, and new CSS3 features such as shadows, opacity, and rounded corners.
3. Gain a thorough understanding of JavaScript programming to create dynamic and interactive web pages. Focus on syntax, control structures, object-oriented features, and practical applications like game development and DOM manipulation.
4. Acquire skills to set up a Node.js development environment and utilize Node.js modules and NPM. Implement server-side functionality, including file system operations and database connectivity using MongoDB.
5. Develop the ability to create robust web applications using Express.js, focusing on middleware usage. Handle various types of requests and responses, and integrate a basic React.js front-end with RESTful APIs to build a complete web application.

Teaching-Learning Process (General Instructions)

1. These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.
2. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
3. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, focused listening, and formulating questions.
4. To make ***Critical thinking***, ask at least three higher order Thinking questions in the class.
5. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
6. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them
7. Discuss how some ***concept can be applied to the real world*** - and when that's possible.

UNIT – I	**07 Hours**

Markup Language (HTML5): Introduction to HTML and HTML5 - Formatting and Fonts - Commenting Code – Anchors – Backgrounds – Images – Hyperlinks – Lists – Tables – HTML Forms,

Audio, Video Tag **(Text Book 1- Chapter-2)**	
UNIT – II	**08 Hours**
CSS3: Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; Background images, Conflict Resolution, CSS Box Model .CSS3 features: Box Shadow, Opacity, Rounded corners, Attribute selector. **(Text Book 1- Chapter-3)**	
UNIT – III	**08 Hours**
JavaScript: Overview of JavaScript; Object orientation and JavaScript; General syntactic, characteristics; Primitives, operations, and expressions; Screen output and keyboard input. Control statements; Arrays; Functions, Constructors; A brief introduction on pattern matching using regular expressions, DOM Events, JSON **,** Case Study -game development **(Text Book 1- Chapter-4,5)**	
UNIT – IV	**08 Hours**
Node JS: Introduction to NodeJS, Set up Dev Environment, Node JS Modules, Node Package Manager, File System, Events, Database connectivity using Mongo DB. **(Text Book 2- Chapter-1,Chapter 8.2,8.3,8.4)**	
UNIT – V	**08 Hours**
Express.JS: Introducing Express: Basics of Express, Express JS Middleware: Serving Static Pages, React JS basic, REST API, Your first React Web Application. **(Text Book 2- Chapter-7.1,7.4)**	

Course Outcome	Description	Bloom's Taxonomy Level
1	Use the common HTML5 elements(tags) to **develop the** static web pages.	**L3**
2	**Make use of** Cascading Style Sheets with HTML5 elements for visual presentation and design well-structured web pages.	**L3**
3	**Implement** the JavaScript programming concepts to develop client-side scripts and display the contents dynamically.	**L3**

4	**Develop** dynamic server-side applications by employing Node.js event-driven, non-blocking I/O model and its integration with Mongo dB to work with dynamic schemas.	**L3**
5	Utilize basic concepts React.js and Express.js framework, to **construct** REST web applications.	**L3**

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	2	2	2		3				1				2	
CO2	2	2	2		3				1				2	
CO3	2	2	2		3				2				2	
CO4	2	2	2		3				2				2	
CO5	2	2	2		3				2				2	

TEXT BOOKS:

1. Robert W. Sebesta , ”Programming the World Wide Web”, 7th Edition, Pearson Education, 2008.
2. Basarat Ali Syed,” Beginning Node.js “,Apress ,2014

REFERENCES:

1. Lionel Lopez,React Quickstart Step-by-Step Guide to Learning React Javascript Library
2. Kirupa Chinnathambi, JavaScript Absolute Beginner’s Guide, 1st Edition, 2017.
3. Robert W Sebesta, Pearson, Programming the World Wide Web, 7th Edition, 2013.
4. Kirupa Chinnathambi, Learning React, 1 Edition, Addison-Wesley Professional
5. Mark Pilgrim,HTML5Up and Running,O’Reilly, 1st Edition, 2012.

E-Resources:

1. https://www.edureka.co/blog/ebook/web-development-ebook
2. MOOC:https://www.coursera.org/learn/server-side-javascript-with-nodejs

Activity Based Learning (Suggested Activities in Class)

1. Hands-on Lab exercises
2. Mini Project
3. Freecode camp/OpenSource Online Coding Boot Camp
4. Coding platform Challenges – Hackerrank, CodeChef
5. Case based learning

****

DESIGN THINKING AND IDEATION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Subject Code	:	25CT2306	**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P-J	:	1–0–0-2			

Course Learning Objectives:

This Course will enable students to:

6. **Analyze** the historical and cultural context of design thinking.
7. **Generate** innovative ideas using various ideation techniques.
8. **Develop** an empathetic and user-centric mindset.
9. **Analyze** and evaluate case studies and real-world examples of successful design thinking and idea generation projects to extract insights and best practices**.**

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

9. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
10. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
11. Show ***Video/animation*** films to explain functioning of various concepts.
12. Encourage ***Collaborative*** (Group Learning) Learning in the class.
13. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
14. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
15. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
16. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**07 Hours**

INTRODUCTION:

Introduction to Design Thinking, Stages, Understanding the Mindsets-Empathy, Optimism, Embrace Ambiguity, Make it, Learn from Failure, Iterate, Create Confidence, Creativity Convergent & Divergent Thinking.

METHODS AND TOOLS FOR EMPATHIZE PHASE: Methods of Empathize phase: Ask 5 Why/ 5W +H questions, Stakeholders map, Empathy Map, Peer observation, Trend analysis. **Case study:How to** apply Empathize tool and create a design for solving real world problems.
UNIT – II **07 Hours**
METHODS AND TOOLS FOR DEFINE PHASE: Methods of Define Phase: Storytelling, Critical items diagram, Define success. **Case study:** How to leverage methods & tools in the define phase for real world applications. **Methods and Tools for Ideate Phase:** Ideate- Brainstorming, 2x2 matrix, NABC method **Activity: Generate innovative idea and present based** on the methods/ tools learnt so far.
UNIT – III **07 Hours**
Methods and Tools for Prototype & Test phase: **Protypes**- Types of prototypes, Methods of prototyping- focused experiment, Exploration map, Minimum Viable Product, Methods of testing, Feedback capture grid. A/B testing **Activity:** Iterate and improve the idea(s) using the prototype and testing -methods & tools.
UNIT – IV **05 Hours**
Create a pitch, plan for scaling up, Road map for implementation. Design Thinking to Business Process modelling.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Utilize** the core concepts of design thinking stages, including Mindsets- Empathy, Optimism, Embrace Ambiguity, Make it, Learn from Failure, Iterate, Create Confidence, Creativity Convergent & Divergent Thinking to showcase the application of empathize phase methods and tools.	L2
2	**Implement** define tools to frame the problem statement and Ideate tools to generate idea for solving the problem.	L3
3	**Develop** proficiency in prototyping and testing as essential steps in creating successful and user-centered products or solutions.	L6

4	**Apply** effective pitching skills and roadmap for business process modelling.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3	3	3							2		3
CO2	3	3	3	3	3	3			3		3	2		3
CO3	3	3	3	3	3	3			3		3	2		3
CO4	3	3	3	3	3	3			3		3	2		3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. IdrisMootee," Design Thinking for Strategic Innovation",2013, John Wiley & Sons Inc

REFERENCE BOOKS:

1. Brenda Laurel, "Design Research methods and perspectives", MIT press 2003
2. Terwiesch, C. & Ulrich, K.T., 2009. "Innovation Tournaments: creating and identifying Exceptional Opportunities", Harvard business press.
3. Ulrich &Eppinger, "Product Design and Development", 3rd Edition, McGraw Hill, 2004
4. Stuart Pugh, "Total Design: Integrated Methods for Successful Product Engineering, BjarkiHallgrimsson, Prototyping and model making for product design", 2012, Laurence King Publishing Ltd

E-Resources:

1. https://www.mygreatlearning.com/academy/learn-for-free/courses/design-thinking
2. https://www.edx.org/learn/design-thinking
3. https://www.coursera.org/learn/design-thinking-innovation
4. https://alcorfund.com/insight/ideation-tools-purpose-methods/
5. https://onlinecourses.swayam2.ac.in/aic19_de04/preview

****

INNOVATIVE BUSINESSES & BREAKTHROUGH TECHNOLOGIES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Subject Code	:	25CT2307	**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P-J	:	1–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. To equip students with the skills and knowledge needed to apply breakthrough technologies strategically in business innovation.
2. To enable students to comprehend the nature, impact, and potential of breakthrough technologies

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

CONTENTS	**13 HRS**
The User Innovation Paradigm, Going for User Solutions Instead of User Needs – the Lead User Method, How Patents Discourage Innovation User Innovation Communities, Exploring New Combinations of Customer Needs, Determine Users' Needs Ethnographically and Develop Solutions MIT Media Lab Approach, Design and Manufacture of Mass Customized Products with Toolkits/Platforms for User Innovation	
Hands-on Component	**13 HRS**
1. Brainstorming as a Concept Generation Method	

2. Case Studies on Problem-Solving about Analogical Thinking		
Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Discuss innovative solutions by combining diverse customer needs in producing products or services with unique prepositions.	L2
2	Utilize design product tool kits or platforms to the production of customized products that meet individual user preferences and demands.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	3	3	3	2								3
CO2	2	2	3	3	3	2								3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. MIT Open Course Ware – How to Develop Breakthrough Products and Services

REFERENCE BOOKS:

1. Eric Von Hippel, Democratizing Innovation, The MIT Press, February 2006
2. Eric Von Hippel, Stefan Thomke and Mary Sonnack, Creating Breakthroughs at 3M, Harvard Business Review
3. Glen L. Urban, John R. Hauser, "Listening In" to Find & Explore New Combinations of Customer Needs, Journal of Marketing, Sage Journals, 2004

E-Resources:

1. https://ocw.mit.edu/courses/sloan-school-of-management/15-356-how-to-develop-breakthrough-products-and-services-spring-2012/index.htm

****

R FOR DATA ANALYSIS AND APPLICATION DEVELOPMENT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT2308	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Acquire** the concept of data science, data analysis.
2. **Develop** the data analysis graphs and charts using data visualization tools and machine learning with statistics.
3. **Acquire** the basics of R, simulation and usage of built in functions, file processing and graphics programming in applications
4. **Understand** and **acquire** skill on T-Testing and linear optimization designing.
5. **Develop** confidence and efficient writing of R scripts.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**5 Hours**

INTRODUCTION: how to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes.
(Text Book-1: Chapter 1: 1.1 to 1.4, 1.6, Chapter 2: 2.1,2.2, 2.4)

UNIT – II	**5 Hours**

R Programming Structure: Control Statements, Loops, - Looping Over Non-Vector Sets, If-Else, Arithmetic and Boolean Operators,

Default Values for Argument, Return Values, Functions with No Pointers in R, Recursion, Sorting and Searching ***(Text Book-1: Chapter 3: 3.1 to 3.3).***	
UNIT – III	**5 Hours**
Plots and Graphs: Graphics, Creating Graphs, The Workhorse of R Base Graphics, the plot () Function Customizing Graphs,Saving Graphs to Files. (***Text Book-2: Chapter 8: 8.1 tow 8.8)***	
UNIT-IV	**5 Hours**
T-Testing: Probability Distributions, Normal Distribution- Binomial Distribution- Poisson Distributions, BasicStatistics, Correlation and Covariance, T-Tests, -ANOVA. ***(Text Book-2: Chapter 10: 10.1 to 10.6)***	
UNIVT-V	**6 Hours**
ML Model Implementation: Linear Models, Simple Linear Regression and Multiple Regression, Generalized Linear Models, Nonlinear Models, Splines- Decision- Random Forests. (***Text Book-2: Chapter 12: 12.1 to 12.9)***	

Course Outcomes:

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

1. **Understand** the fundamentals, standards of Functions and capabilities of R- Language to **develop**

   data analytics applications.
2. **Formulate** Learning the basic R-Language Constructs and **apply** it to **build** basic educationalapplications.
3. **Demonstrate** the simulation in R-Language, Math functions and files Processing to read and analysis health care data.
4. **Apply** the Principals of Graphics and R-Base Graphics to plot the real time health care applicationsdata.
5. **Develop** basic health care applications and Performing T-Testing, and **Design** and **build** Linear optimization of health care data by using R-Language concepts.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	-	-	-	3	2	-	-	2	2	-		-	3
CO2	2	-	2	-	2	-	-	-	-	3	-		-	-
CO3	2	-	-	-	-	-	-	-	-	3	-		-	2
CO4	2	-	1	-	3	-	-	-	-	3	-		-	-
CO5	3	-	-	-	2	-	-	-	3	3	-		-	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. **The Art of R Programming, Norman Matloff, Cengage Learning: Efficient R Programming: A Practical Guide to Smarter Programming 1st Edition - Colin Gillespie & Robin Lovelace- First Edition.**
2. **Jared P. Lander, R for Everyone: Advanced A nalytics and Graphics, Second Edition 2017.**

REFERENCE BOOKS:

1. **R Cookbook, PaulTeetor, Oreilly: R Cookbook [R CKBK] [Paperback] R Cookbook[RKBK][Paperback]Mar 31, 2011 by Paul Teetor.**
2. **R in Action,Rob Kabacoff, Manning: R in Action: Data Analysis and Graphics withRNov 5,2018 | Unabridged by Robert Kabacoff and Dale Ogden**

E-Resources:

1. http://nptel.ac.in/courses
2. https://www.listendata.com/p/r-programming-tutorials.html#:~:text=Hands-on%20R%20Programming%20Tutorials%201%20Best%20R%20Tutorials,R%204%20Shiny%20Tutorials%205%20R%20Interview%20Questions
   http://agce.sets.edu.in/cse/ebook/DBMS%20BY%20RAGHU%20RAMAKRISHNAN.pdf

Activity Based Learning (Suggested Activities in Class)

1. Data analysis and data exploration using group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

****

PRODUCT MARKETING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Subject Code	:	25CT2309	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. To put the students on the cutting edge of knowledge in making strong and actionable recommendations to managements in the field of marketing.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
   Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I:	**26 Hours**

Introduction to Product Marketing, Visual representation of Product Marketing, building a Customer Persona, Product-Market Fit, Product Life Cycle, Product Marketing vs Regular Marketing, Responsibilities of a Product Marketer, Critical steps of Product Marketing, Measuring the impact of Product Marketing.

Resources:

Video: https://olympus.mygreatlearning.com/courses/91026/pages/introduction-to-product-marketing?module_item_id=3776737

Course Outcomes:

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

1. Develop a deep understanding of product marketing fundamentals, including its role in bridging product development and market demand.

2. Analyze data performance and make data-driven decisions to optimize marketing strategies and achieve better results.

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3	3				3							2
CO2				3	2					2				2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

REFERENCE BOOKS:

1. Philip Kotler, Kevin Lane Keller, Abraham Koshy and Mithileswar Jha, Marketing Management, Pearson, 14th Edition, 2013.

E-Resources:

1. https://learning.productmarketingalliance.com/on-demand/advanced-product-marketing-certified
2. https://learning.productmarketingalliance.com/leadership-on-demand/product-marketing-certified-leadership#course-overview

Activity Based Learning (Suggested Activities in Class)

1. Real-world case studies of successful and unsuccessful product launches.
2. Brainstorm different messaging strategies for various customer segments.

****

PROBABILITY AND STATISTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	:	25CT2401	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Apply** statistical principles and probability concepts to solve complex problems in real-world scenarios involving uncertainty and randomness.
2. **Evaluate** and select appropriate probability distributions and statistical techniques to analyze and interpret data accurately in various applications.
3. **Justify** the use of estimation methods and hypothesis testing techniques for drawing meaningful inferences about population parameters.
4. **Analyze** and interpret sample test results for different statistical relationships, such as means, variances, correlation coefficients, regression coefficients, goodness of fit, and independence, to make informed decisions.
5. **Identify** sample tests using appropriate statistical procedures to investigate the significance of observed data and communicate findings effectively.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I : Probability	**09 Hours**

Definitions of Probability, Addition Theorem, Conditional Probability, Multiplication Theorem, Bayes' Theorem of Probability	
UNIT – II: Random Variables and their Properties and Probability Distributions	**09 Hours**
Discrete Random Variable, Continuous Random Variable, Joint Probability Distributions Their Properties, Probability Distributions: Discrete Distributions: Binomial, Poisson Distributions and their Properties; Continuous Distributions: Exponential ,Normal, Distributions and their Properties.	
UNIT – III: Estimation and testing of hypothesis	**06 Hours**
Sample, Populations, Statistic, Parameter, Sampling Distribution, Standard Error, Un-Biasedness, Efficiency, Maximum Likelihood Estimator, Notion & Interval Estimation.	
UNIT – IV: Sample Tests-1	**07 Hours**
Large Sample Tests Based on Normal Distribution , Small Sample Tests : Testing Equality of Means, Testing Equality of Variances, Test of Correlation Coefficient	
UNIT – V: Sample Tests-2	**08 Hours**
Test for Regression Coefficient; Coefficient of Association, 2 – Test for Goodness of Fit, Test for Independence.	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Apply** the principles of probability to solve complex problems in various real-world scenarios.	L2 & L3
2	**Solve** and **compare** different probability distributions, including discrete and continuous random variables, in order to make informed decisions and predictions.	L2 & L3

3	**Apply** statistical estimation techniques, such as maximum likelihood estimation and interval estimation, to draw meaningful inferences about population parameters from sample data.	L3
4	**Examine** hypothesis testing methods, including large and small sample tests, to assess the significance of observed data and draw valid conclusions.	L4
5	**Analyze** statistical relationships and perform sample tests to assess the Equality of means in different populations, Correlation coefficients between variables to determine the strength and direction of the relationship. Independence of variables using appropriate statistical tests to assess the absence of any relationship.	L4

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	2		2				1				
CO2	3	2	2		2				1				
CO3	3	2	2						1				
CO4	3	2	2		2				1				
CO5	3	2	2		2				1				

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. Probability & Statistics for Engineers and Scientists, Walpole, Myers, Myers, Ye. Pearson Education.

REFERENCE BOOKS:

1. Probability, Statistics and Random Processes T. Veerarajan Tata McGraw – Hill
2. Probability & Statistics with Reliability, Queuing and Computer Applications, Kishor S. Trivedi, Prentice Hall of India ,1999

E-Resources:

1. https://nptel.ac.in/courses/106104233
2. https://nptel.ac.in/courses/117103067
3. https://nptel.ac.in/courses/103106120
4. https://www.coursera.org/learn/probability-intro#syllabus
5. https://nptel.ac.in/courses/111104073

Activity Based Learning (Suggested Activities in Class)

1. Tools like Python programming, R programming can be used which helps student to develop a skill to analyze the problem and providing solution.
2. Regular Chapter wise assignments/ Activity/Case studies can help students to have critical thinking, developing an expert mind set, problem-solving and teamwork.

Following are Assignments/ Activities Can be carried out using either R programming language or Python Programming or excel solver.

1. There are n people gathered in a room. What is the probability that at least 2 of them will have the same birthday? (Use excel solver, R Programming, Python Programming)
   a. Use simulation to estimate this for various n., and Produce Simulation Graph.
   b. Find the smallest value of n for which the probability of a match is greater than 0.5.
   c. Explore how the number of trials in the simulation affects the variability of our estimates.

2. **Case Study 1: Customer Arrivals at a Coffee Shop**
    a. A coffee shop wants to analyze the number of customer arrivals during its morning rush hour (7:00 AM to 9:00 AM). The shop has been recording the number of customer arrivals every 15 minutes for the past month.
    b. Data: The data consists of the number of customer arrivals recorded at the coffee shop during each 15-minute interval for the past month.
    c. Here is a sample of the data:

Time Interval	**stomer Arrivals**
00 AM - 7:15 AM	**6**
15 AM - 7:30 AM	**4**
30 AM - 7:45 AM	**9**
45 AM - 8:00 AM	**7**
00 AM - 8:15 AM	**5**
15 AM - 8:30 AM	**8**
30 AM - 8:45 AM	**10**
45 AM - 9:00 AM	**6**

analyze the customer arrivals and determine the probability distribution that best fits the data. Specifically, explore both discrete and continuous probability distributions, including the binomial, Poisson, exponential, and normal distributions.

3. **Case Study 2: Comparing the Performance of Two Groups**
    a. Suppose you are a data analyst working for a company that manufactures a new energy drink. The marketing team conducted a promotional campaign in two different cities (City A and City B) to determine the effectiveness of the campaign in increasing sales. The sales data for a random sample of customers in each city was collected over a week. Your task is to compare the average sales between the two cities and test whether there is a significant difference in the variance of sales.
    b. **Data**: Let's assume the following sample data for the number of energy drinks sold in each city:

City A: [30, 28, 32, 29, 31, 33, 34, 28, 30, 32]

City B: [25, 24, 26, 23, 22, 27, 29, 30, 26, 24]

perform a two-sample t-test to test the equality of means and a test for equality of variances using Python's SciPy library.

4. **Case study 3:** testing independence between two categorical variables.
   a. Data: Sample of 100 employees, and each employee is classified as either Male or Female. They were asked to rate their job satisfaction on a scale of 1 to 5, where 1 represents low satisfaction and 5 represents high satisfaction. The data is as follows:

Employee	**Gender**	**Job Satisfaction**
1	Male	4
2	Female	3
3	Male	2
4	Female	5
...	...	...
100	Female	4

   b. Test for independence between gender and job satisfaction, use the chi-squared test in R.

****

DESIGN AND ANALYSIS OF ALGORITHMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	:	25CT2402	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39(L) + 26(P) Hours
L–T–P-J	:	3–0–2-0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the non-recursive and recursive algorithms and to represent efficiency of these algorithms in terms of the standard Asymptotic notations.
2. **Devise** the Brute Force and Divide and Conquer techniques to design the algorithms and apply these methods in designing algorithms to solve a given problem.
3. **Explain** the Decrease and Conquer, Transform and Conquer algorithm design techniques, and Time versus Space Trade-offs.
4. **Get the idea** of Greedy method and dynamic programming methods and apply these methods in designing algorithms to solve a given problem.
5. **Describe** and illustrate the idea of Backtracking and Branch and Bound algorithm design techniques to solve a given problem.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	08 Hours
INTRODUCTION: What is an Algorithm? Fundamentals of Algorithmic Problem Solving. ***(Text Book-1: Chapter 1: 1.1 to 1.2)*** **FUNDAMENTALS OF THE ALGORITHMS EFFICIENCY:** Analysis Framework, Asymptotic Notations and Standard notations and common functions ***(Text Book-2: Chapter 3: 3.1, 3.2)***, Mathematical Analysis of Non-recursive and Recursive Algorithms, ***(Text Book-1: Chapter 2: 2.1, 2.3, 2.4,)***	
UNIT – II	**08 Hours**
BRUTE FORCE: Background, Selection Sort, Brute-Force String Matching. ***(Text Book-1: Chapter 3: 3.1, 3.2)*** **DIVIDE AND CONQUER:** General method, Recurrences: The recursion-tree method, The master method. ***(Text Book-2: Chapter 4: 4.4, 4.5***), Merge sort, Quick sort, Binary Search, Multiplication of large integers, **Case study:** Strassen's Matrix Multiplication. ***(Text Book-1: Chapter 4: 4.1 to 4.3, 4.5)***	
UNIT – III	**06 Hours**
DECREASE & CONQUER: General method, Insertion Sort, Graph algorithms: Depth First Search, Breadth First Search, Topological Sorting **TRANSFORM AND CONQUER:** **Case study:** Heaps and Heap sort. **TIME AND SPACE TRADEOFFS:** Input Enhancement in String Matching: Horspool's algorithm, Hashing: Open and Closed hashing. ***(Text Book-1: Chapter 5: 5.1 to 5.3, Chapter 6: 6.3 to 6.4, Chapter 7: 7.2 to 7.3)***	
UNIT – IV	**09 Hours**
GREEDY TECHNIQUE: General method of Greedy technique, Single-Source Shortest Paths: General method, The Bellman-Ford algorithm, Single-Source Shortest Paths in DAGs, Dijkstra's Algorithm ***(Text Book-2: Chapter 24: 24.1 to 24.3).*** Minimum Spanning Trees: Prim's Algorithm, Optimal Tree problem: Huffman Trees;	

Case study: Kruskal's Algorithm. ***(Text Book-1: Chapter 9: 9.1, 9.2, 9.4).*** **DYNAMIC PROGRAMMING:** General method, The Floyd-Warshall Algorithm, Johnson's algorithm for sparse graphs ***(Text Book-2: Chapter 25: 25.1 to 25.3)***, The Knapsack problem ***(Text Book-1: Chapter 8: 8.4).***

UNIT – V	**08 Hours**
LIMITATIONS OF ALGORITHMIC POWER P, NP and NP-complete problems ***(Text Book-1: Chapter 11: 11.3***) **BACKTRACKING:** General method, N-Queens problem, Subset-sum problem. ***(Text Book-1: Chapter 12: 12.1***) **BRANCH AND BOUND:** General method, Travelling Salesman problem, Approximation algorithms for TSP. **Case study:** Knapsack Problem. ***(Text Book-1: Chapter 12: 12.2, 12.3***)	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Analyze the given recursive and non-recursive algorithms for time using step count, substitution method, recurrence tree method and mathematical analysis method. Represent the complexity of the algorithm using asymptotic notation.	L3
2	Solve sorting, searching, matrix multiplication problems based on divide and conquer design technique and implement in C programming language.	L3

3	Apply algorithms for graph-based problems (DFS, BFS and Topological sorting) using decrease and conquer design techniques. Distinguish the trade-offs between space and time complexity.	L3
4	Apply algorithms for finding the shortest path and minimum spanning tree for a given graph using greedy and dynamic programming techniques and implement the 0/1 knapsack problem in C programming language.	L3
5	Apply an efficient algorithm to solve N-Queens problem, Subset-sum problem, Knapsack and Traveling salesman problem using branch and bound and backtracking design technique. Describe the limitation of algorithmic power in terms of P, NP, NP hard and NP complete categorizations.	L3
6	Implement the graph-based algorithms including DFS, BFS, Warshall's Algorithm, Floyd's Algorithm, Kruskal's Algorithm, and Dijkstra's Algorithm in C programming language. (**Lab Experiments**)	L3

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	2	3	-	-	-	-	-	-	-	1	-	2	3	
CO2	2		-	-	3	-	-	-	-	1	-	2	3	
CO3	2	2	3	3	-	-	-	-	-	1	-	2	2	
CO4	2	2	3	-	3	-	-	-	-	1	-	2	3	
CO5	2	2	3	-	-	-	-	-	-	1	-	2	2	
CO6	2	2	3	-	3	-	-	-	-	1	-	2	3	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Anany Levitin, "Introduction to the Design & Analysis of Algorithms", 2nd Edition, Pearson Education, 2011.
2. Thomas H. Cormen, Charles E.Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, PHI, 2014.

REFERENCE BOOKS:

1. Horowitz E., Sahni S., Rajasekaran S, "Computer Algorithms", Galgotia Publications, 2001.
2. R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, "Introduction to the Design and Analysis of Algorithms: A Strategic Approach", Tata McGraw Hill, 2005.

E-Resources:

1. https://nptel.ac.in/courses/106/101/106101060/
2. http://cse01-iiith.vlabs.ac.in/
3. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
4. https://www.coursera.org/specializations/algorithms

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solutions to a problem through programming.

DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY

Total Contact Hours: 26

Following are experiments to be carried out using either C programming language or Object-oriented programming language:

1. Implementation of Binary Search algorithm.

2. Sort a given set of n integer elements using the Merge Sort method and compute its time complexity. Demonstrate this algorithm using the Divide-and-Conquer method.
3. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Demonstrate this algorithm using the Divide-and-Conquer method.
4. Incorporate the array data structure and demonstrate whether a given unweighted graph is connected or not using the DFS method.
5. Implement the graph traversal technique using the BFS method to print all the nodes reachable from a given starting node in an unweighted graph.
6. Compute the Transitive Closure for a given directed graph using Warshall's algorithm.
7. For a given weighted graph, construct an All-Pairs Shortest Paths problem using Floyd's algorithm and implement this algorithm to find the shortest distance and their shortest paths for every pair of vertices.
8. Implement 0/1 Knapsack problem using Dynamic Programming Memory Functions technique
9. Find Minimum Cost Spanning Tree for a given weighted graph using Prim's and Kruskal's algorithm.
10. From a given vertex in a weighted connected graph, determine the Single Source Shortest Paths using Dijkstra's algorithm.
11. Implement N-Queens problem using Backtracking technique.
12. Case Study

****

DATABASE MANAGEMENT SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	:	25CT2403	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39 (L)+ 26(P) Hours
L–T–P-J	:	3–0–2-0			

Course Learning Objectives:

This Course will enable students to:

1. Acquire the concept of databases and Relational Model., Entity-Relationship Model and relational model for creating and designing databases for the real-world scenario.
2. Develop queries to extract data from the databases using a structured query language.
3. Draw ER Diagrams and Optimize the Database design using Normalization Concepts.
4. Understand the importance of Transaction Management, Concurrency control mechanism and recovery techniques.
5. Understand NoSQL Database concepts and Demonstrate CRUD operations on MongoDB.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

7. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
8. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
9. Show ***Video/animation*** films to explain functioning of various concepts.
10. Encourage ***Collaborative*** (Group Learning) Learning in the class.
11. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
12. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
13. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
14. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

RELATIONAL DATABASES

Purpose of Database System – Views of data – Data Models – Database System Architecture- Database System Applications - Introduction to relational databases – Structure of Relational Databases – Database Schema – Keys – Schema Diagrams.

(Text Book 1: Chapter 1, 2)

UNIT – II	**08 Hours**

RELATIONAL QUERY LANGUAGE

Overview of the SQL Query Language - SQL Data Definition - Basic Structure of SQL Queries - Additional Basic Operations - Aggregate Functions - Nested Subqueries - Join Expressions – Views – Transactions - SQL Data Types and Schemas – Authorization - Accessing SQL from a Programming Language - Functions and Procedures – Triggers.

(Text Book 1: Chapter 3, 4, 5)

UNIT – III	**06 Hours**

DATABASE DESIGN

Entity-Relationship model – E-R Diagrams – Complex Attribute – Mapping Cardinalities – ER-to-Relational Mapping – Features of Good Relational Designs - Decomposition Using Functional Dependencies – Non-
loss Decomposition – First, Second, Third Normal Forms, Dependency Preservation – Boyce/Codd Normal Form – Multi-valued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form

(Text Book 1: Chapter 6, 7)

UNIT – IV	**09 Hours**

TRANSACTION MANAGEMENT

Transaction Concepts – ACID Properties – A simple Transaction Model - Transaction Atomicity and Durability - Transaction Isolation - Schedules – Serializability – Transactions as SQL Statements - Concurrency control –Lock Based Protocols - Deadlock Handling – Recovery System - Failure Classification - Recovery Algorithm - Buffer Management - ARIES

(Text Book 1: Chapter 17, 18, 19)

UNIT – V	**08 Hours**

NOSQL Databases
NOSQL Databases: Introduction – CAP Theorem – Document-Based NOSQL Systems and MongoDB – CRUD Operations - NOSQL Key-Value Stores - Column-Based or Wide Column NOSQL Systems - NOSQL Graph Databases and Neo4j
(Text Book 2 : Chapter 24)

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand the basic concepts of database management system and Relational Model	L2
2	Utilize SQL concepts to build and manipulate relational databases for a given schema.	L3
3	Create E-R diagrams and design relational schema, Apply normalization techniques in designing the relational database	L3
4	Understand the Transaction Management, concurrency control and recovery management techniques.	L2
5	Use NoSQL queries and build databases in MongoDB for the given collection.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	1	-	-	-	-	-	-	-	-	-	-		-	2
CO2	1	2	3	-	3	-	-	-	2	1	-		-	3
CO3	1	1	2	-	-	-	-	-	2	1	-		-	3

CO4	**1**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**		**-**	**1**
CO5	**1**	**2**	**3**	**-**	**3**	**-**	**-**	**-**	**2**	**1**	**-**		**-**	**3**

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, "Database System Concepts", Seve nth Edition, McGraw Hill, 2020.
2. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Seventh Edi tion, Pearson Education, 2017

REFERENCE BOOKS:

1. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", Third Edition, McGraw-Hill, 2003.
2. C.J. Date, A. Kannan, S. Swamynatham: "An Introduction to Database Systems", Eight Edition, Pearson Education, 2012.

E-Resources:

1. https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-what-is-database-management-system
2. **https://www.mongodb.com/resources/basics/database-management-system**
3. **https://www.oracle.com/in/database/what-is-database/**
4. **https://onlinecourses.swayam2.ac.in/cec19_cs05/preview**
5. **https://onlinecourses.nptel.ac.in/noc19_cs46/preview**

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solution to a problem through programming.

Design and analysis of algorithms Laboratory

Total Contact Hours: 26

1. Database Table Creation and Manipulation - Create a database table and add constraints such as primary key, unique, check, and not null. Insert, update, and delete rows using SQL DDL and DML commands.
2. Foreign Keys and Referential Integrity - Create a set of tables with foreign key constraints to ensure referential integrity between them.
3. Querying with Conditions and Aggregates - Query the database tables using various WHERE clause conditions and implement aggregate functions like SUM, AVG, COUNT, etc.
4. Subqueries and Basic Joins - Perform queries involving subqueries and simple join operations to retrieve data from multiple tables.
5. Advanced Join Operations - Explore and implement natural joins, equijoins, and outer joins in your queries.
6. UserDefined Functions and Stored Procedures - Write and execute userdefined functions and stored procedures in SQL for various operations.
7. Transactions and Control Commands - Execute complex transactions and understand the use of Data Control Language (DCL) and Transaction Control Language (TCL) commands.
8. SQL Triggers - Write SQL triggers for handling insert, delete, and update operations on a database table.
9. Views - Create views to enhance manageability.
10. NoSQL Database Operations - Create and manage document based data using MongoDB database tools.

Open ended Questions

1. GUIBased Database Application - Develop a simple GUIbased database application incorporating features such as table creation, constraints, queries, joins, functions, procedures, transactions, and triggers.
2. Case Study: RealLife Database Application

****

INTRODUCTION TO ARTIFICIAL INTELLIGENCE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	:	25CT2404	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P-J	:	3–0–0-0			

Course Learning Objectives:

This Course will enable students to:

1. To explore introductory survey of concepts and techniques in artificial intelligence.
2. To learn about with methods for search, classification, reasoning and machine learning.
3. Familiar with applications including core AI (game, Planning), robotics , computer vision and natural language understanding

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION

What is AI? Foundations of artificial intelligence (AI). History of AI; The State of the Art. Agents and Environments, Good Behavior, The Nature of Environments, The Structure of

Agents, Problem- solving agents, Example problems, searching for solutions, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Heuristic Functions ***(Text Book 1: Chapter 1 – 1.1-1.4, Chapter 2 – 2.1-2.4, Chapter 3 – 3.1-3.6)***	
UNIT – II	**08 Hours**
ROBOTICS AND CLASSICAL PLANNING Classical Planning: Definition, Algorithms for Planning as State-Space Search, Planning Graphs, Other Classical Planning Approaches, Analysis of Planning Approaches, Robotics: Introduction, Robot Hardware, Robotic Perception, planning to move, Planning Uncertain Movements. ***(Text Book 2: Chapter 11 – 11.1-11.7 Chapter 26 – 26.1-26.6, Chapter 4 – 4.4.1, 4.4.2, Chapter 6 – 6.3, C12- 12.2, Chapter 13- 13.3.1, 13.3.2, Text Book 1: Chapter 9 – 9.1, 9.2.1, 9.3)***	
UNIT – III	**08 Hours**
Uncertainty, Naive Bayes and Probabilistic Reasoning Acting Under Uncertainty, Review of Basic Probability, Bayes Theorem, Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Exact Inference in Bayesian Networks, Time and Uncertainty, Inference in Temporal Models ***(Text Book 2: Chapter 12.1–12.6 Chapter 13 – 13.1-13.3 Chapter 14-14.1-14.2)***	
UNIT – IV	**08 Hours**
LEARNING Forms of Learning, Supervised Learning, Learning Decision Trees, Evaluating and Choosing the Best Hypothesis, Regression and Classification with Linear Models, Artificial Neural Networks. Reinforcement Learning ***(Text Book 1: Chapter 19 – 19.1-19.6)***	
UNIT – V	**07 Hours**
APPLICATIONS Natural Language Processing: Language Models, Text Classification, Information Retrieval, Information Extraction, Computer Vision: Image Formation, Early Image-Processing Operations, Object Recognition by Appearance, Reconstructing the 3D World, Object Recognition from Structural Information, Using Vision. ***(Text Book 1: Chapter* 23- 23.1 Chapter 25-25.1-25.6)**	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Demonstrate fundamental understanding of artificial intelligence (AI) especially the notion of problem-solving using AI techniques, current scope and limitations thereof.	L2
2	Apply basic principles of AI in solutions that require problem solving, knowledge representation, and learning	L3
3	Demonstrate awareness and a fundamental understanding of applying AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	1	2												3
CO2	3	2												3
CO3	2	2												3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Stuart Russell and Peter Norvig. Artificial Intelligence – A Modern Approach, Pearson Education Press, 4th Edition.

REFERENCE BOOKS:

1. Artificial Intelligence by Elaine Rich, Kevin Knight and Shivashankar B Nair, Tata McGraw Hill.

2. Introduction to Artificial Intelligence and Expert Systems by Dan W. Patterson, Pearson Education.

E-Resources:

1. https://nptel.ac.in/courses/106105079
2. https://www.coursera.org/learn/introduction-to-ai

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solutions to a problem through programming.

****

COMPUTER ORGANIZATION AND ARCHITECTURE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	: 25CT2405	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–J	: 3–0–0–0		

Course Learning Objectives:

This Course will enable students to:

1. Understand the Architecture and programming of ARM microprocessor.
2. Develop program using Arm instruction set and appreciate the advanced features provided in the ARM
3. Understand the exception handling techniques.
4. Study in detail the concept of instruction level parallelism and concepts of pipelining.
5. Understand various cache memory mapping techniques and memory Organization.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

15. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
16. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
17. Show ***Video/animation*** films to explain functioning of various concepts.
18. Encourage ***Collaborative*** (Group Learning) Learning in the class.
19. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
20. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**05 Hours**

An Overview of Computing Systems:
History of Computers, The Computing Device.
The ARM7TDMI Programmers' Model:
Introduction, Data types, Processor Modes, Registers, Program Status Registers, The vector Table.
Assembler Rules and Directives: Structure of Assembly Language Modules, Registers, Directives and Macros.
Loads, Stores and Addressing: LODS and STORES instructions, Operand Addressing, ENDIANNES

Text Book-1: 1.1 to 1.3; 2.1 to 2.3; 4; 5.3, 5.4, 5.5	
UNIT – II	**05 Hours**
Constants and Literal Pools: The ARM Rotation Scheme, Loading Constants and address into Registers **Logic and Arithmetic:** Flags and their Use, Compare instructions, Data Processing Instructions **Loops and Branches:** Branching, Looping, Conditional Execution, Straight-Line Coding **Subroutines and Stacks:** Stack, Subroutines, Passing parameters to subroutines. **Text Book-1: 6.1 to 6.4; 7.1 to 7.4; 8.2 to 8.5; 13.1 to 13.4**	
UNIT – III	**05 Hours**
Mixing C and Assembly Language: Inline Assembler Embedded Assembler, Calling Between C and Assembly. **Exception Handling:** Interrupts, Error Conditions, Processor Exception Sequence, The Vector Table, Exception Handlers, Exception Priorities, Procedures for Handling Exceptions. **Text Book-1: 18.1 to 18.4; 14.1 to 14.8**	
UNIT – IV	**12 Hours**
Pipelining: Basic and Intermediate Concepts: Introduction, The Major Hurdle of Pipelining, How Pipelining Implemented, what makes Pipelining hard to Implement, Extending the MIPS Pipeline to Handle Multicycle Operations, The MIPS R4000 Pipeline, Crosscutting Issues, score boarding Technique. **Text Book-2: C.1 to C.7**	
UNIT – V	**12 Hours**
Memory Hierarchy: Introduction, Cache Performance, Six basic cache Optimizations, Virtual Memory, Protection and examples of Virtual Memory, Fallacies and Pitfalls. **Text Book-2: B.1 to B.6**	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		

1	**Interpret** the knowledge of the internal architecture and organization of ARM microprocessors to understand their components and functionalities.	L2
2	**Apply** the instruction set of ARM Microprocessor by writing Assembly language programs.	L3
3	**Understand** the various exception handling techniques.	L2
4	**Demonstrate** the integration of C and Assembly language to manage function calls between C and Assembly language.	L2
5	**Apply** the concept of instruction-level parallelism and understand the principles of Pipelining techniques.	L3
6	**Understand** memory hierarchy and its impact on computer cost/performance.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3		2										2	
CO2	3		3		1								2	
CO3	3	1	1										2	
CO4	3	1	1										2	
CO5	3	2	1										2	
CO6	3	2	1										2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. William Hohl, “ARM Assembly Language”, 2nd Edition, CRC Press, 2009.

2. John L Hennessy, David A Patterson, “Computer Architecture, A Quantitative Approach”, 6th Edition, Morgan Kaufmann publishers, 2019.

REFERENCE BOOKS:

1. David A Patterson, John L Hennessy, “Computer Organization and Design”, 4th Edition, Morgan Kaufmann publishers, 2010.
2. Steve Furber, “ARM System-on-chip Architecture”, 2nd Edition, Pearson Publications, 2000.
3. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, “Computer Organization”, 5th Edition, Tata McGraw Hill, 2002.

E-Resources:

1. https://www.udemy.com/topic/arm-cortex-m/
2. https://www.edx.org/school/armeducation
3. https://onlinecourses.nptel.ac.in/noc22_cs93/preview

Activity Based Learning (Suggested Activities in Class)

1. Mini project implementation using Assembly Language Programming.
2. Demonstration of solution to a problem through programming.

****

SMALL E-BUSINESS LAUNCH

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	:	25CT2406	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Experience** with a set of tools and methods for setting up an online service.
2. **Knowledge** of the role of multiple functions in creating a small business.
3. **Ability** to coordinate multiple, interdisciplinary tasks to achieve a common objective.
4. **Emphasis** of specific knowledge from other courses through practice and reflection in an action-oriented setting.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**06 Hours**

INTRODUCTION:

Introduction, Identifying Customer Needs, Opportunity Identification & Planning, Idea Generation and Validation, Entrepreneurial Culture, Digital Business Opportunity

(Text Book-1: Chapter 1: 1.1, 1.4 to 1.9, 1.11 to 1.17, 1.22 to 1.28)

UNIT – II	**06 Hours**

Business Plan & its Components, Business Model and Strategies, Selection of Appropriate Business Model, Business Model Canvas, Feasibility

(Text Book-1: Chapter 3: 3.1 to 3.18)	
UNIT – III	**06 Hours**
Lean thinking, Specifications, Wireframe Generation and Validation. ***(Text Book-1: Chapter 5: 5.1 to 5.4, Chapter 5: 5.11 to 5.19)***	
UNIT – IV	**08 Hours**
Service-Market Fit, legal Aspects & Entrepreneurial Law, Logistics of goods delivery, Virtual Trade Shows ***(Text Book-1: Chapter 7: 7.2 to 7.17***)	
Course Outcomes: At the end of the course the student will be able to: **1.** Identify customer needs and pain points**.** **2.** Generate concept and validate idea**.** **3.** Use design thinking to create a wire frame of your online service**.** **4.** Launch your app/website. **5.** Register your small business.	
List of Laboratory/Practical Experiments activities to be conducted (if any):	**13 Hrs**
Building of Business Models and seminars on deployment tools	

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	3												3
CO2	3	2	2											2
CO3	2	2	2											2
CO4	3	2	2											3
CO5	3	2												3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Fisher, Steven, Duane, Ja-Nae (2016). Start-up Equation – A Visual Guidebook for Building your Start-up, New Delhi: Tata McGraw-Hill
2. Ulrich, K. T. (2019). Product design and development. Tata McGraw-Hill Education, 7th edition

REFERENCE BOOKS:

1. Holt H., David (2017). Entrepreneurship: New Venture Creation, New Delhi: Pearson Education
2. Zimmerer, W. Thomas and Scarborough, M. Norman and Doug Wilson (2009). Essentials of Entrepreneurship and Small Business Management, 5/e; New Delhi: Prentice Hall India

Activity Based Learning (Suggested Activities in Class)

1. Proposal Handout & Presentation
   a. The Problem
   b. Need
   c. Existing Alternatives
   d. Potential Market

2. Wireframe Sketches, Target Specifications
   a. List Critical Customer Needs
   b. For a., Prepare a List of Target Specifications
   c. Prepare Wireframe

****

IDEA GENERATION & VALIDATION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	:	25CT2407	**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P-J	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Generate, develop, and pitch innovative new venture ideas
2. Evaluate the attractiveness and feasibility of new venture opportunities
3. Analyze industries, markets, and competitors

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

Lecture method means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.

1. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
2. Show ***Video/animation*** films to explain functioning of various concepts.
3. Encourage ***Collaborative*** (Group Learning) Learning in the class.
4. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
5. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
6. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
7. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

CONTENT	**13 Hours**
Process of Idea Generation; Problem Identification, Idea formation, Idea Generation Techniques: Brainstorming, Mind Mapping, Story Boarding, Attribute Listing, Idea Validation and Risk Assessment	

List of Laboratory/Practical Experiments activities to be conducted (if any):	**13 HRS**

1. Get out of the Building Exercise – Meet at least 20 persons in your targeted segment and understand their pain points and propose ideas that may solve their problems.
2. Discuss with mentors and zero in on one idea.
3. Make a hand drawn functional sketch of the proposed solution

4. Idea Validation with potential customers - Meet at least 20 customers and validate the proposed idea with them.
5. Prepare a comprehensive report, highlighting the learnings and the implications on the proposed solution and pivots (if any).

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Classify idea generation techniques to validate ideas.	L2
2	Interpret to pitch innovations on new venture ideas.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	3	3	3									3
CO2	2	2	3	3	3									3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Kotter, J.P. and Whitehead, L.A. (2010). Buy-In: Saving Your Good Idea from Getting Shot Down.

REFERENCE BOOKS:

1. Belsky, S. (2010). Making Ideas Happen: Overcoming the Obstacles between Vision and Reality.

****

Technical writing

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	:	25CT2408	**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P-J	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Teach how to improve writing skills and level of readability
2. Tell about what to write in each section
3. Summarize the skills needed when writing a Title
4. Infer the skills needed when writing the Conclusion
5. Ensure the quality of paper at very first-time submission

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

Lecture method means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.

1. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
2. Show ***Video/animation*** films to explain functioning of various concepts.
3. Encourage ***Collaborative*** (Group Learning) Learning in the class.
4. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
5. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
6. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
7. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

–	UNIT I	6 Hours

INTRODUCTION TO RESEARCH PAPER WRITING:

Planning and Preparation, Word Order, Breaking up long sentences, Structuring Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness

UNIT-II	6 Hours

PRESENTATION SKILLS:

Clarifying Who Did What, Highlighting Your Findings, Hedging and Criticizing, Paraphrasing and Plagiarism, Sections of a Paper, Abstracts, Introduction

UNIT-III	5 Hours
TITLE WRITING SKILLS: Key skills are needed when writing a Title, key skills are needed when writing an Abstract, key skills are needed when writing an Introduction, skills needed when writing a Review of the Literature, Methods, Results, Discussion, Conclusions, The Final Check	
UNIT-IV	**5 Hours**
RESULT WRITING SKILLS: Skills are needed when writing the Methods, skills needed when writing the Results, skills are needed when writing the Discussion, skills are needed when writing the Conclusions	
UNIT-V	**5 Hours**
VERIFICATION SKILLS: Useful phrases, checking Plagiarism, how to ensure paper is as good as it could possibly be the first-time submission	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand that how to improve your writing skills and level of readability	L2
2	Learn about what to write in each section	L2
3	Understand the skills needed when writing a Title	L2
4	Understand the skills needed when writing the Conclusion	L2
5	Ensure the good quality of paper at very first-time submission	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	2	2	2								2	
CO2	2	2	2	2	2								2	

CO3	2	2	2	2	2								2	
CO4	2	2	2	2	2								2	
CO5	2	2	2	2	2								2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011
2. Day R How to Write and Publish a Scientific Paper, Cambridge University Press 2006
3. Goldbort R Writing for Science, Yale University Press (available on Google Books) 2006
4. Highman N, Handbook of Writing for the Mathematical Sciences, SIAM. Highman's book 1998.

REFERENCE BOOKS:

1. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011

****

INTRODUCTION TO IPR AND REGISTRATION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	: 25CT2409		**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. To understand the concepts IPR
2. To understand Trademarks, Trade Secretes and GI of goods.
3. To understand Copyrights, Patents and Industrial Designs.
4. To learn about how to manage IP rights and legal aspects.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**04 Hours**

INTRODUCTION TO IPR:

Introduction to Intellectual Property Rights, types of intellectual property, importance of intellectual property rights, Evolution of IP acts and treaties, Agencies responsible for IPR registrations, Role and value of IP in international commerce, Issues affecting IP internationally. ***(Text Book-1: Chapter 1).***

NIT – II	**07 Hours**

Trade Marks:
Purpose and function of trademarks, Acquisition of trade mark rights, transfer of rights, Selecting and evaluating trademark, registration of trademarks, claims.
(Text Book-1: Chapter 2,3,4).
Trade Secrets: Trade secret law, determination of trade secret status, liability for misappropriation of trade secrets, trade secret litigation.
(Text Book-1: Chapter 22).

UNIT – III	07 Hours

COPYRIGHTS:
Fundamentals Of Copyright Law, Originality of Material, Right of Reproduction, Right to Perform the Work Publicly, Copyright Ownership Issues, Notice of Copyright. (Text Book-1: Chapter 10,11).
PATENTS: Foundation of Patent Law, Patent Searching Process, Basic Criteria of Patentability Industrial Designs: Kind of Protection Provided in Industrial Design. (Text Book-1: Chapter 17,18).
Managing IP Rights:
Acquiring IP Rights: letters of instruction, joint collaboration agreement, Protecting IP Rights: non-disclosure agreement, cease and desist letter, settlement memorandum. Transferring IP Rights: Assignment contract, license agreement, deed of assignment.
(Text Book-1: Chapter 19).

UNIT – IV	08 Hours

DESIGN-
Meaning, Definition, Object, Registration of Design, Cancellation of Registration, International convention on design, functions of Design. Semiconductor Integrated circuits and layout design Act- 2000.
(Text Book-2: Chapter 8).
BASIC TENENTS OF INFORMATION TECHNOLOGY ACT-2000 –
IT Act - Introduction
E-Commerce and legal provisions
E- Governance and legal provisions
Digital signature and Electronic Signature.
(Text Book-2: Chapter 12).

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Demonstrate** the basic concepts of IPR and their role and value internationally.	L2
2	**Interpret** trademark, their values and trade secrets for liability and litigations.	L3

3	**Devise** knowledge on copyrights, patents and how to manage and protect IP.	L3
4	**Examine** convention on design and basic information technology act.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	3	3	3									3
CO2	3	3	3	3	3	3			3		3			3
CO3	3	3	3	3	3	3			3		3			3
CO4	3	3	3	3	3	3			3		3			3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Deborah Bouchoux ,"Intellectual Property: The Lawof Trademarks, Copyrights, Patents, and Trade Secrets", 4th Edition, Delmar Cengage Learning.
2. Elizabeth Verkey," Intellectual Property: Law and Practice", 2015 Edition, Eastern Book Company

REFERENCE BOOKS:

1. Kompal Bansal , Parikshit Bansal, "Fundamentals of Intellectual Property for Engineers", 2014, BS Publications.
2. Prabuddha Ganguli," Intellectual property right - Unleashing the knowledge economy", Ganguli, 1st edition (1 July 2017), McGraw Hill Education.

E-Resources:

1. Inventing the Future: An introduction to Patents for small and medium sized Enterprises; WIPO publication No. 917. URL: www.wipo.int/ebookshop.

2. Looking Good : An Introduction to Industrial Designs for Small and Medium-sized Enterprises; WIPO publication No.498. URL: www.wipo.int/ebookshop.
3. Creative Expression: An Introduction to Copyright and Related Rights for Small and Medium-sized Enterprises; WIPO publication No. 918. URL: www.wipo.int/ebookshop
4. Making a Mark: An Introduction to Trademarks for Small and Medium-sized Enterprises; WIPO publication No. 900. URL: www.wipo.int/ebookshop

****

DevOps FUNDAMENTALS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER –IV

Course Code	:	25CT2410	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–J	:	1–0–0–2			

Course Learning Objectives:

This course will enable students to:

1. **To understand** the principles and lifecycle of DevOps and its benefits in modern software development.
2. **To describe** the fundamentals of Git and its role in version control, including common workflows and collaborative practices.
3. **To outline** the concepts of CI/CD, the setup of Jenkins, and the basics of Docker containerization and orchestration.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I: Introduction to DevOps:	**06 Hours**
Understanding DevOps, Benefits, DevOps Lifecycle ***(Text Book-1: Chapter 1)*** **Version Control with** Git Git Fundamentals, Workflow, Collaboration with Git ***(Text Book-1: Chapter 4)***	
UNIT – II: Continuous Integration and Continuous Deployment (CI/CD)	**10 Hours**
CI/CD Concepts, Jenkins, Creating and managing CI/CD pipelines with code. ***(Text Book-1: Chapter 5)***	

UNIT – III: Containerization with Docker	10 Hours
Introduction to Containers, Learning Docker fundamentals, containers, and images., Exploring container orchestration with Kubernetes. ***(Text Book-1: Chapter 8)***	

Course Outcomes:

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

1. **Apply** Git for version control by creating repositories, managing branches, and collaborating on code using workflows.
2. **Construct** automated CI/CD pipelines using Jenkins and manage them using pipeline-as-code practices.
3. **Analyze** the role of Docker and Kubernetes in containerization and orchestration for scalable and reliable deployment.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	2	2	2	2	-	-	-	2	2	2		-	2
CO2	2	2	2	2	2	-	-	-	2	2	2		-	2
CO3	2	2	2	2	2	-	-	-	2	2	2		-	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. **"DevOps Automation Cookbook" by Ekambar Kumar Singirikonda**

E-Resources:

1. https://www.ebooks.com/en-us/book/211369550/devops-automation-cookbook/ekambar-kumar-singirikonda/
2. https://www.mygreatlearning.com/academy/learn-for-free/courses/introduction-to-devops1
3. https://intellipaat.com/academy/course/devops-free-course/

Activity Based Learning (Suggested Activities in Class)

1. Development of project using DevOps tools and group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

****

THEORY OF COMPUTATION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	: 25CT3501	**Credits**	:	04
Hours / Week	:04 Hours	**Total Hours**	:	39 Hours (Theory)+13Hours (Tutorial)
L–T–P-J	:3–1–0-0			

Course Learning Objectives:

This Course will enable students to:

6. **Understand** the Automata Theory and Formal Languages to build efficient design of FA
7. **Identify** Regular Expression and recognize the properties that make a language regular and construct the FA of the language.
8. **Devise** the technique to minimize DFA and understand the importance of minimization in optimizing automata for efficient language recognition.
9. **Get the idea** to Interpret and design different PDA for a given language
10. **Describe** the finite automata and formal languages, equipping them with the knowledge and skills necessary to analyse and design TM for language recognition tasks.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

21. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
22. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
23. Show ***Video/animation*** films to explain functioning of various concepts.
24. Encourage ***Collaborative*** (Group Learning) Learning in the class.
25. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
26. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
27. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
28. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION TO FINITE AUTOMATA:

Study and Central Concepts of Automata Theory, Finite Automata -Yet Another Method for Defining Languages, Deterministic and Nondeterministic Finite Automata, Finite Automata with Epsilon – transitions. An Application: Text Search.
(Text Book-1: Chapter 1: 1.1, 1.7, Chapter 2: 2.2 to 2.5)
(Text Book-2: Chapter 5: Page no: 52)

UNIT – II	**08 Hours**

REGULAR EXPRESSIONS AND LANGUAGES:
Regular Expressions, Finite Automata and Regular Expressions, Algebraic Laws of Regular expressions, Applications of Regular Expressions, Properties of Regular Languages - Pumping Lemma, Applications of the Pumping Lemma, Closure Properties of Regular languages, Equivalence and minimization of Automata.
(Text Book-1: Chapter 3: 3.1 to 3.4, Chapter 4: 4.1, 4.2, 4.4)

UNIT – III	**09 Hours**

CONTEXT – FREE GRAMMARS AND LANGUAGES:
Context-Free Grammars, Parse Trees, Ambiguity in Grammars and Languages, Properties of Context free languages-Normal Forms of Context-Free Grammars, The Pumping Lemma for Context Free Languages, Closure Properties of Context-Free Languages.
(Text Book-1: Chapter 5: 5.1 to 5.2.3, 5.4, Chapter 7: 7.1 to 7.3)

UNIT – IV	**06 Hours**

PUSHDOWN AUTOMATA:
Definition of the Pushdown automation (PDA), The Language of PDA, Equivalence of PDA's and CFG's-From Grammars to Push Down Automata and PDA to Grammars, Deterministic Pushdown Automata
(Text Book-1: Chapter 6: 6.1 to 6.4)

UNIT – V	**08 Hours**

INTRODUCTION TO TURING MACHINE:
Problems that Computers Cannot Solve, The Turing Machine, Programming Techniques for Turing Machine,Extensions to the Basic Turing Machine.
(Text Book-1: Chapter 8 : 8.1, 8.4)

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Utilize the basic concepts of formal languages of finite automata techniques such as DFA, NFA and E-NFA	L3
2	Develop Finite Automata for different Regular Expressions and Languages and minimization of Finite Automata to Regular Expression.	L3

3	Analyze context-free grammars, ambiguity and Chomsky normal form grammars to design computer languages	L4
4	Construct context free, regular, Interpret and design different PDA for a given language	L5
5	Design Turing machine to solve problems.	L6

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	3											3
CO2	3	3	3											3
CO3	1	1		3	1									2
CO4	2	2	3	3	1									3
CO5	1	2	3	3	3									3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. J.E. Hopcroft, R. Motwani, and J. D. Ullman, Introduction to Automata Theory, Languages and Computation, 3rd Edn. Pearson Education , New Delhi 2008
2. Daniel I. A. Cohen, Introduction to Computer Theory, 2nd Edition, Wiley IndiaStudent Edition, 2008.

REFERENCE BOOKS:

1. K.L.P. Misra and N. Chandrashekaran. Theory of Computer Science- Automata, Languages and Computation, 3rd Edn. PHI, New Delhi, 2007
2. C. Martin - Introduction to Languages and the Theory of Computation 2ndEdn, TMH, New Delhi, 2000.

Activity Based Learning (Suggested Activities in Class)

3. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
4. Demonstration of solution to a problem through programming.

List of Laboratory/Practical Experiments activities to be conducted.
1. Design a Program for creating machine that accepts three consecutive one.
2. Design a Program for creating machine that accepts the string always ending with 101.
3. Design a Program for Mode 3 Machine
4. Design a program for accepting decimal number divisible by 2.
5. Design a program for creating a machine which accepts string having equal no. of 1's and 0's.
6. Design a program for creating a machine which count number of 1's and 0's in a given string.
7. Design a Program to find 2's complement of a given binary number.
8. Design a Program which will increment the given binary number by 1.
9. Design a Program to convert NDFA to DFA.
10. Design a Program to create PDA machine that accept the well-formed parenthesis.
11. Design a PDA to accept WCWR where w is any string and WR is reverse of that string and C is a Special symbol.
12. Design a Turing machine that's accepts the following language an b n c n where n>0.

****

SECURE SOFTWARE ENGINEERING AND PROJECT MANAGEMENT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT3502	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

1. **Outline** software engineering principles, activities and different process models involved in building large software programs. Identify ethical and professional issues and explain why they are of concern to Software Engineers.
2. **Explain** the role of Agile development and Extreme Programming (XP).
3. **Devise** the process of requirement gathering, requirement classification, requirement specification and requirements validation.
4. **Recognize** the importance Project Management with its methodologies.
5. **Identify** software quality parameters and quantify software using measurements and metrics.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

29. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
30. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
31. Show ***Video/animation*** films to explain functioning of various concepts.
32. Encourage ***Collaborative*** (Group Learning) Learning in the class.
33. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
34. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
35. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
36. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

Introduction to Software Engineering: Software Engineering, The software Process, The software Engineering practice

Process Models: A generic process model, Prescriptive process models, Waterfall model, Incremental process models, Evolutionary process models, Concurrent
models, Specialized process models. SSDLC and CIA Model.

Textbook 1: Chapter 1: 1.1 to 1.7 Chapter 2: 2.1 to 2.4

UNIT – II	**08 Hours**

Agile Development: What is Agility?, Agility and the cost of change. What is an agile Process?, Extreme Programming (XP), Other Agile Process Models, A tool set for Agile process

Principles that guide practice: Software Engineering Knowledge, Core principles, Principles that guide each framework activity

Basics of DevOps and DevSecOps

Textbook 1: Chapter 3: 3.1 to 3.6, Chapter 4: 4.1 to 4.3

UNIT – III	08 Hours

Understanding Requirements: Requirements Engineering, Eliciting Requirements, Developing use cases, Building the requirements model, Validating Requirements, Requirement Analysis. STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege), abuse/misuse cases.
Design: Design Concepts, Understanding Architectural Design, Component Level Design, UI/UX design, Webapp Design and Secured CODING (CERT C, OWASP).
Testing: UNIT, Integration, System, Acceptance Testing, Black box/WhiteBox Testing, Secured Testing(SAST, DAST, Penetration testing).

Textbook 1: Chapter 5: 5.1 to 5.7, Chapter 6: 6.1 to 6.5
Case study: Software Requirement specification

UNIT – IV	08 Hours

Introduction to Project Management:
Introduction, Project and Importance of Project Management, Contract Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, some ways of categorizing Software Projects, Stakeholders, Setting Objectives, Business Case, Project Success and Failure, Risk Management, PMM, CMM, Software Cost & Effort Management.

Case study- **W5HHH Principles, CMMI Levels.**
Textbook 2: Chapter 1: 1.1 to 1.14

UNIT – V	07 Hours

Software Quality:
Defining software quality, quality models, ISO 9126, product and process metrics, product versus process quality management, Quality Management systems, process capability models, techniques to enhance software quality, testing-types, Software reliability, quality plans. **SQA**

Case study: **Writing Test cases on real time projects.**
Textbook 2: Chapter 13: (13.1 to 13.11)

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Understand** the activities involved in software engineering and analyze the role of various process models	L1 & L2
2	**Explain** various software testing methods and to understand the importance of agile methodology and DevOps	L2
3	**Describe** the basics of object-oriented concepts and build a suitable class model using modelling techniques	L2
4	**Illustrate** the role of project planning and quality management in software	L2 & L3

	development	
5	**Interpret** the importance of activity planning and different planning models.	L2 & L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2										2			3
CO2	2										2			3
CO3	3			2	3						2			3
CO4	3	2		3	3						3			3
CO5	3	2		3	2						2			3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill.
2. Bob Hughes, Mike Cotterell, Rajib Mall: Software Project Management, 6th Edition, McGraw Hill Education, 2018.

REFERENCE BOOKS:

1. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India.
2. Software Engineering, by Ian Sommerville 10th edition, Pearson global edition
3. Software Engineering, an Engineering approach-JamesF. Peters, Witold Percy, John Wiley
4. Software Security: Building Security In" by Gary McGraw
5. Agile Application Security: Enabling Security in a Continuous Delivery Pipeline by Jim Bird, et al
6. Threat Modeling: Designing for Security" by Adam Shostack

E-Resources:

1. https://onlinecourses.nptel.ac.in/noc20_cs68/preview

2. https://www.youtube.com/watch?v=WxkP5KR_Emk&list=PLrjkTql3jnm9b5nr-ggx7Pt1G4UAHeFlJ
3. http://elearning.vtu.ac.in/econtent/CSE.php
4. http://elearning.vtu.ac.in/econtent/courses/video/CSE/15CS42.html

https://nptel.ac.in/courses/128/106/128106012/ (DevOps)

Activity Based Learning (Suggested Activities in Class)

5. Demonstration of solution to a problem using different process models like V model.
6. Real world problem solving and its cost estimation using COCOMO models.
7. Project management activity- Project scheduling using Tools for near real time projects.
8. Testing activity-Writing Test case & Test plan for the real time projects.

****

MACHINE LEARNING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT3503	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39(L) + 26(P)
L–T–P-J	:	3–0–2-0			

Course Learning Objectives:

This Course will enable students to:

1. Understand the basic theory underlying machine learning algorithms.
2. Formulate machine learning problems corresponding to different applications.
3. Understand how to evaluate models generated from data.
4. Apply the algorithms to a real problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION

Introduction to Machine Learning, Types of ML, Applications of ML.

(Text Book-1: Chapter 1: 1.1 to 1.2)

MATHEMATICS FOR MACHINE LEARNING
Bayes' Theorem, Gaussian Distribution, Data, Models and Learning, Empirical Risk Minimization, Parameter Estimation, Probabilistic Modeling and Inference.
(Reference Book-1: Chapter 6: 6.3, 6.5, Chapter 8: 8.1 to 8.4)

UNIT – II	08 Hours

REGRESSION
Overview of Supervised Learning-Introduction, Variable Types and Terminology, Model Selection and the Bias–Variance Trade-off, Regression – Linear, Non-Linear, Univariate, Multivariate

CLASSIFICATION
Classification - Logistic Regression, Naïve Bayes, K Nearest Neighbour, SVM, Decision Tree

(Text Book 2: Chapter 2 – 2.1, 2.2, 2.9, Chapter 3 – 3.2, 3.2.3, Chapter 4 – 4.4.1, 4.4.2, Chapter 6 – 6.3, C12- 12.2, Chapter 13- 13.3.1, 13.3.2, Text Book 1: Chapter 9 – 9.1, 9.2.1, 9.3)

UNIT – III	08 Hours

CLUSTERING
Cluster Analysis - Proximity Matrices, Dissimilarities Based on Attributes, Clustering Algorithms, K-means, Gaussian Mixtures as Soft K-means Clustering, Hierarchical Clustering,
(Text Book 2: Chapter14)

ENSEMBLE METHODS
Ensemble – Voting, Bagging, Boosting
(Text Book 1: Chapter 17 – 17.4, 17.6, 17.7)

AdaBoosting, Gradient Boosting, Random Forest,
(Text Book 2: Chapter 10 – 10.4, 10.10.2, Chapter 15 – 15.1, 15.2, 15.3)

UNIT – IV	08 Hours

REINFORCEMENT LEARNING
Introduction, Model-Based Learning, Temporal Difference Learning.
(Text Book 1: Chapter 18 – 18.1, 18.4, 18.5)

DIMENSIONALITY REDUCTION
Introduction. Subset Selection, PCA, Singular Value Decomposition and Matrix Factorization, Multidimensional Scaling, Linear Discriminant Analysis
(Text Book 1:Chapter 6 – 6.1, 6.2, 6.3, 6.6, 6.7, 6.8)

UNIT – V	07 Hours

DESIGN AND ANALYSIS OF MACHINE LEARNING EXPERIMENTS
Introduction, basic principles of experimental design, Guidelines for Machine Learning Experiments, Cross-Validation and Resampling Methods, Measuring Classifier Performance, Interval Estimation, Hypothesis Testing, ANOVA.

***(Text Book 1: Chapter* 19 – 19.1, 19.4, 19.5, 19.6, 19.7, 19.8, 19.9, 19.12)**

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand types of learning algorithms. Utilize foundational concepts in probability to apply machine learning effectively.	L3
2	Develop solutions for regression and classification problems using supervised learning algorithms with a given dataset.	L3
3	Apply clustering algorithms and ensemble methods to solve problems.	L3
4	Understand Reinforcement Learning and to optimize the learning model using feature engineering, dimensionality reduction and PCA.	L3
5	Understand how to evaluate the performance of machine learning algorithms	L2
6	Develop models in python to solve complex problems using ML algorithms and analyze the model performance.	L4

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	1	2											3	
CO2	3	2											3	
CO3	2	2											3	
CO4	3	2											3	
CO5	2	2											2	
CO6	3	3	3	2	3				3	3	2		3	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Ethem Alpaydin, Introduction to Machine Learning (Adaptive Computation and Machine Learning series), The MIT Press, Third Edition.
2. Trevor Hastie, Robert Tibshirani, Jerome H. Friedman, The Elements of Statistical Learning, Second Edition, Springer.

REFERENCE BOOKS:

1. Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, Mathematics for Machine Learning, Cambridge University Press.
2. Thomas M. Mitchell, Machine Learning, McGraw- Hill, Inc. New York

E-Resources:

1. https://www.kaggle.com/learn/intro-to-machine-learning
2. https://learn.microsoft.com/en-us/training/modules/introduction-to-machine-learning/
3. **https://onlinecourses.nptel.ac.in/noc22_cs29/preview**

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solutions to a problem through programming.

MACHINE LEARNING LABORATORY

Total Contact Hours: 26

Following are experiments to be carried out using Python:

1. Data Collection and Preprocessing
2. Exploratory Data Analytics.
3. Univariate and Multivariate Linear Regression
4. Logistic Regression
5. Decision Tree Classification
6. Naïve Bayes Classification
7. K Nearest Neighbour Classification
8. SVM
9. K Means Clustering
10. Random Forest
11. XGBoost

****

PRODUCT DESIGN & DEVELOPMENT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	: 25CT3504		**Credits**	:	03
Hours / Week	:	02 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** and understand customer pain points.
2. **Devise** the best possible solution to build PoC to test in agile way.
3. **Explain** the different roles and responsibilities in product development
4. **Get the idea** of tools and technologies with automation in building product.
5. **Describe** and illustrate the collaborative product development.

Teaching-Learning Process (General Instructions)

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, by following Idea to PoC cycle.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**
Introduction, product development process & organization, product planning, opportunity identification, identifying customer needs ***(Text Book-1: Chapter 1 - 5)***	
UNIT – II	**08 Hours**
Product specifications, concept - generations, selection, testing, product architecture ***(Text Book-1: Chapter 6 - 10)***	
UNIT – III	**07 Hours**
Industrial design, design for environment, design for manufacturing & supply chain ***(Text Book-1: Chapter 11 - 13)***	
UNIT – IV	**08 Hours**

Prototyping, robust design, patents & intellectual property, service design ***(Text Book-1: Chapter 14 - 17).***	
UNIT – V	**08 Hours**
Product development economics, project management ***(Text Book-1: Chapter 18 - 19***)	

Course Outcomes:

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

1. **Describe** the Product design and development with exposure with a set of tools and methods.
2. **Explain** on coordination with cross disciplinary teams to achieve a common objective with exposure on collaborative development in team(s).
3. **Describe** the role of multiple functions (e.g. strategy, marketing, engineering, legal, customer service, finance, supply chain, operations, etc.) in creating a new product.
4. **Identify the problem statement and explain** the product development life cycle by building PoC.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3												3
CO2	3	3	2											3
CO3	3	3												3
CO4	3	3	2											3
CO5	3	3												3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PROJECT DELIVERABLES 13 hrs

1. Proposal Handout & Presentation
    1. The Problem
    2. Need
    3. Existing Alternatives
    4. Potential Market
2. PoC / Prototype, Target Specifications
    1. Core solution
    2. Resources (Environment, Automation, Time to Market)

3. Build PoC / Prototype

3. Demonstrate PoC including Workflow automation
4. Plan / Register your business (Optional)

TEXT BOOKS:

1. Ulrich, K. and Eppinger, S.; Product Design and Development | 7th Edition, McGraw Hill; ISBN - 13 : 978-9390113231 (https://pdfhost.io/v/jcYs4cRR8_Karl_T_Ulrich_Steven_D_Eppinger_Maria_C_Yang_Product_Design_and_Development_2019_1pdf.pdf)

REFERENCE BOOKS:

1. Christensen, C.M.; The Innovator's Dilemma; Harvard Business Review Press; ISBN: 978-1-63-369178-0
2. Chesbrough, H. W.; Open Innovation; Harvard Business Review Press; ISBN: 978-1-42-210283-1

E-Resources:

1. https://onlinecourses.swayam2.ac.in/imb19_mg01/preview

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving as a team building soft skills as well as getting exposure to product development skills and automation as appropriate.
2. Demonstration of PoC

****

OPERATING SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT3505	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39(T)+26(P) Hours
L–T–P–J	:	3–0–2–0			

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the basic concepts and functions of operating systems.
2. **Understand** Processes and Threads
3. **Analyze** Scheduling algorithms.
4. **Understand** the concept of Deadlocks.
5. **Analyze** various Memory and Virtual memory management, File system and storage techniques.
6. **Discuss** the goals and principles of protection in a modern computer system.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

9. **Lecture method** means it includes not only traditional lecture method, but different type of teaching methods may be adopted to develop the course outcomes.
10. **Interactive Teaching:** Adopt the Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
11. Show **Video/animation** films to explain functioning of various concepts.
12. Encourage **Collaborative** (Group Learning) Learning in the class.
13. To make **Critical thinking**, ask at least three Higher order Thinking questions in the class.
14. Adopt **Problem Based Learning**, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
15. Show the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.

UNIT – I	**08 Hours**

OS Overview and System Structure: Introduction to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating

System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Computing environments. **Operating System Services:** User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; ***Textbook – 1: 1.1 -1.10,1.12,2.1-2.8.***	
UNIT – II	**08 Hours**
Process Management: Process concept; Process scheduling; Operations on processes. **Multi- threaded Programming:** Overview; Multithreading models; Threading issues. **Process Scheduling**: Basic concepts; Scheduling Criteria; Scheduling Algorithms. ***Textbook – 1: 3.1-3.3,4.1,4.2,4.4,5.1-5.3***	
UNIT – III	**08 Hours**
Process Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors Deadlocks. **Deadlocks:** System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. ***Textbook-1: 6.1-6.7,7.1-7.7***	
UNIT – IV	**07 Hours**
Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. **Virtual Memory Management**: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. ***Textbook-1: 8.1-8.6,9.1-9.6***	
UNIT – V	**08 Hours**
File System, Implementation of File System: File system: File concept; Access methods; Directory structure; File system mounting; File sharing. Protection: Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management. **Mass storage structures:** Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. **Protection and Security:** Protection: Goals of protection, Principles of protection, System Security: The Security Problem, Program Threats, System and Network Threats.	

Textbook-1: 10.1-10.6,12.1-12.6,14.1,15.1-15.3

List of Laboratory/Practical Experiments:		
Exp. No	**Division of Experiments**	**List of Experiments**
1	System Calls	Write a C program to create a new process that exec a new program using system calls fork(), execlp() & wait()
2		Write a C program to display PID and PPID using system calls getpid () & getppid ()
3		Write a C program using I/O system calls open(), read() & write() to copy contents of one file to another file
4	Process Management	Write a C program to implement multithreaded program using pthreads
5		Write C program to simulate the Round Robin CPU scheduling algorithms
6	Process synchronization	Write a C program to simulate producer-consumer problem using semaphores
7	Deadlock	Write a C program to simulate Bankers algorithm for the purpose of deadlock avoidance.
8		
9	Memory Management	Write a C program to simulate FIFO page replacement algorithms
10		
11	I/O System	Write a C program to simulate the single level directory technique
12		Write a C program to simulate the indexed file allocation strategies.

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Illustrate** the key concepts and structures of operating systems, including process,	L2

	memory, and storage management, protection, security, distributed systems, and system calls.	
2	**Apply** concepts of process management, including process scheduling and operations, multi-threaded programming models and issues, and process scheduling criteria and algorithms, resulting in a practical understanding of efficient process handling in operating systems.	L3
3	**Apply** concepts of process synchronization and deadlock management methods for handling, preventing, avoiding, detecting, and recovering from deadlocks.	L3
4	**Compare and contrast** various memory management techniques to enforce memory protection and manage memory allocation and deallocation efficiently.	L2
5	**Demonstrate** the various file management techniques, disk scheduling methods for efficient resource utilization and interpret the system, network, program threats and employ protection principles to safeguard the system resources.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2											2		
CO2	2	3	3	3								2	2	
CO3	3	3	2	2								2	2	
CO4	2	2	3	3								2	3	
CO5	2	2	3	3								2	2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

4. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 8th edition, Wiley-India, 2010

REFERENCE BOOKS:

1. Operating Systems-Internals and Design Principles, William Stallings, 6th Edition, Pearson Education, 2009.

2. Operating Systems: A Modern Perspective, Gary J. Nutt, Addison-Wesley, 1997

E-Resources:

7. **Modern Operating Systems** by Andrew S. Tanenbaum - Known for its comprehensive coverage of modern operating systems principles and design.
8. Operating System Fundamentals - Course (nptel.ac.in)
9. Introduction to Operating Systems - Course (nptel.ac.in)
10. Intro to Operating Systems 1: Virtualization | Coursera
11. Intro to Operating Systems 3: Concurrency | Coursera
12. Intro to Operating Systems 2: Memory Management | Coursera

Activity Based Learning (Suggested Activities in Class)

1. Implementing Operating System concepts by doing Mini projects.
2. Case study to compare various Operating Systems and their performance.
3. Present real-world case studies or problems related to operating systems design and performance optimization, encouraging students to apply theoretical knowledge to analyze and propose solutions.

****

INTERACTIVE ART AND CREATIVE CODING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT3506	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

9. The fundamentals of computational thinking applied through p5.js
10. Best practices for designing software within an event-driven, object-oriented, real-time framework
11. How to ideate, conceptualize, and design original creative works in the medium of software

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**7 Hours**

Computational Thinking: Overview, Stages, Cornerstones, Pillars, Expressing and Analyzing Algorithms, Concepts of Computational Thinking, Techniques Associated with Computational Thinking, Computational Thinking in a Classroom.

(Text Book-1: Chapter 1: 1.1 to 1.3, 1.8, Chapter 2: 2.1,2.2, 2.6)

Unit-II	**8 Hours**

p5.js: Overview, Creative Graphics & Interactive Experiences, p5.js Web Editor, p5.js and Processing, Using a Local Server, First Sketch, Color, Constants, DOM, Data, Environment, Events, Foundation, I/O, Image, Lights, Camera, Math, Rendering, Shape.

(Text Book-1: Chapter 3: 3.1 to 3.8, Chapter 4: 4.1 to 4.6)
Unit-III **9 Hours**
p5.js-contd: node.js and socket.io, 3D/WebGL, Coordinate System and Shapes, Structure, Transform, Typography, Anatomy of Functions, Built-in p5.js Functions, Touch, Unit Testing and Test-Driven Development. ***(Text Book-1: Chapter 5: 5.1 to 5.13)***
Unit-IV **8 Hours**
p5.js – Advanced Data, Sound, Mobile – Acceleration Ball Bounce, Simple Draw, Acceleration Color, Shake Ball Bounce, Titled 3D Box, p5.js libraries, Linking p5.js to HTML file. ***(Text Book-2: Chapter 2: 2.1 to 2.11)***
Unit-V **7 Hours**
p5.ble.js – Overview, Introduction to Arduino and Bluetooth, Usage, p5.ble.js using Arduino, Bluetooth Le and p5.ble p5.bots – Overview, APIs, Servers, Functions p5.clickable – Overview, Creating, Moving & Resizing a Clickable, Clickable Events. ***(Text Book-2: Chapter 4: 4.1 to 4.9)***
List of Laboratory/Practical Experiments activities to be conducted (if any): Download and install sketch.js and setup a sketch to handle mouse/touch input. Setup a sketch to spawn particles with your mouse. Use three.js to setup a screen with a spinning cube. Create a web-based VR application made with three.js How do you update an object's transformation using three.js? Design an animation system using three.js

Course Outcomes: At the end of the course the student will be able to:		
1	Examine fundamentals of computational thinking	
2	Write scripts using p5.js	
3	Ideate, conceptualize, and design original creative works	
4	Analyze and Critique Interactive projects	
5	Understand the programming fundamentals of functions	

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3	3								2		2	1
CO2	3	3	3		3	2	2	1	3				2	1
CO3	3	3	1		3								2	1
CO4			3		3	2							2	1
CO5						3	3	3					2	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Lauren McCarthy, Casey Reas, Ben Fry, Getting Started with p5.js: Making Interactive Graphics in JavaScript and Processing, Maker Media Inc.

2. Karl Beecher, Computational Thinking: A Beginner's Guide to Problem Solving and Programming, The Chartered Institute for IT

REFERENCES BOOKS:

1. Creative Coding and Data Visualization with p5.js", Scott Muray, O'Reilly (13 January 2017)

2. "Processing: Creative Coding and Computational Art", Ira Greenberg Apress; 2nd Printing Edition (June 1, 2007)

****

QUANTUM COMPUTING FUNDAMENTALS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT3507	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0-0			

Course Learning Objectives:

This Course will enable students to:

1. **Explain** the basic concepts of quantum computation and its physics.
2. Use various operators of quantum computation and work on quantum transformation.
3. **Illustrate** the working of some standard quantum algorithms.
4. **Analyze** the complexities involved in working of quantum algorithms.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**07 Hours**

FUNDAMENTAL CONCEPTS:

Global Perspectives, Quantum Bits, Quantum Computation, Quantum Algorithms, Quantum Information, ***(PART I, Chapter : 1.1, 1.2, 1.3, 1.4, 1.6 Pg. No. 01)***

Postulates of Quantum Mechanisms. ***(Chapter 2: 2.2 Pg. No.80)***

UNIT – II	**07 Hours**

QUANTUM COMPUTATION:

Quantum Circuits – Quantum algorithms, Single Orbit operations, Control Operations, Measurement,

Universal Quantum Gates, Simulation of Quantum Systems, ***(PART II, Chapter 4: 4.1, 4.2, 4.3, 4.4, 4.5, 4.7, Pg. No. 171)*** Quantum Fourier transform, Phase estimation, Applications, ***(Chapter 5: 5.1, 5.2, 5.3)*** Quantum search algorithms – Quantum counting– Speeding up the solution of NP – complete problems – Quantum Search for an unstructured database. ***(Chapter 6: 6.1, 6.3, 6.4, 6.5)***
UNIT – III **08 Hours**
QUANTUM COMPUTERS: Guiding Principles, Conditions for Quantum Computation, Harmonic Oscillator Quantum Computer, Optical Photon Quantum Computer – Optical cavity Quantum electrodynamics, Ion traps, Nuclear Magnetic resonance. ***(Chapter 7: 7.1, 7.2, 7.3, 7.4 ,7.5, 7.6, 7.7)***
UNIT – IV **09 Hours**
QUANTUM INFORMATIONS: Quantum noise and Quantum Operations – Classical Noise and Markov Processes, Quantum Operations, Examples of Quantum noise and Quantum Operations – Applications of Quantum operations, Limitations of the Quantum operations formalism, ***(PART III, Chapter 8: 8.1, 8.2, 8.3, 8.4, 8.5 Pg. No. 353)*** Distance Measures for Quantum information. ***(Chapter 9: 9.1, 9.2, 9.3 Pg. No. 399)***
UNIT – V **08 Hours**
QUANTUM ERROR CORRECTION: Introduction, Shor code, Theory of Quantum Error –Correction, Constructing Quantum Codes, Stabilizer codes, Fault – Tolerant Quantum Computation, ***(Chapter 10: 10.1, 10.2, 10.3, 10.4, 10.5, 10.6 Pg. No. 425)*** Entropy and information – Shannon Entropy, Basic properties of Entropy, Von Neumann, Strong Sub Additivity, ***(Chapter 11: 11.1, 11.2, 11.3, 11.4 Pg. No. 500)*** Data Compression, Entanglement as a physical resource***. (Chapter 12: 12.2, 12.5)***

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Explain** basic concepts in Quantum computing.	L1
2	**Illustrate** applications of Quantum computing.	L2
3	**Explain** principles in the design of Quantum Computers.	L1

4	**Analyze** applications and limitations of Quantum operations.	L4
5	**Apply** concepts in Quantum Error Correction.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	1												
CO2	3	1												
CO3	3	1												
CO4	3	2												
CO5	3	2	1											

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Michael A. Nielsen. & Issac L. Chiang, "Quantum Computation and Quantum Information", South Asia Edition 2013, Cambridge University Press, Reprint 2022.

REFERENCE BOOKS:

"Quantum Computing, A Gentle Introduction" , Eleanor G. Rieffel and Wolfgang H. Polak MIT press (2014).

2. Mikio Nakahara and Tetsuo Ohmi, "Quantum Computing", CRC Press (2008). 3. N. David Mermin, "Quantum Computer Science", Cambridge (2007).

3. An introduction to Quantum Computing, Phillip Kaye, Raymond Laflamme, Muchele Mosca, Oxford University Press, 2007, ISBN-13: 978-0198570493, ISBN-10: 019857049X.

E-Resources:

1. https://nptel.ac.in/courses/106106232

2. https://www.youtube.com/watch?v=teraaPiaG8s

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solution to a problem through programming.

****

IoT FUNDAMENTALS: ARCHITECTURE TO ANALYTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	: 25CT3508		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Develop** a solid foundation in IoT concepts and principles.
2. **Acquire** knowledge necessary to design, implement, and manage IoT architectures.
3. **Determine** solutions for troubleshooting communication between smart objects.
4. **Develop** expertise in handling IoT data, applying analytics, and ensuring security and privacy.
5. **Develop** knowledge necessary to troubleshoot communication between smart objects.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**07 Hours**

INTRODUCTION TO IOT:

Introduction, Definition, Physical design of IoT, Logical design of IoT, Application of IoT, IoT enabling technologies, IoT levels & Deployment templates, Domain-specific IoT.

(Text Book-4, chapter-1)

IoT challenges

(Text Book-2, chapter-1)

UNIT – II	**08 Hours**

IOT ARCHITECTURE AND ENGINEERING IOT:
Drivers behind the new network architectures, comparing IoT architecture, a simplified IoT architecture, IoT data management and compute stack.
(Text Book- 2, chapter-2)
Smart objects: The "Things in IoT": Sensors, Actuators, and Smart Objects; Sensor Networks ***(Text Book-2, chapter-3)***

UNIT – III	**07 Hours**

CONNECTING SMART OBJECTS:
Communication Criteria, IoT Access Technologies: Physical and MAC layers, Web Communication Protocols for connected devices, SOAP, REST, HTTP Restful, and Web Sockets. Internet Connectivity Principles: Internet Connectivity, Internet-based communication, Network Layer: IP versions, IP addressing in IoT, Zigbee,6LoWPAN, Routing over Low Power and Lossy Networks.
(Text Book- 2, chapter-4)

UNIT – IV	**09 Hours**

DATA AND ANALYTICS FOR IOT:
An introduction to data analytics for IoT, Machine Learning, Big data analytics tools and technology, Edge streaming analytics, Network analytics.
(Text Book- 2, chapter-7).
Security and privacy in the IoT: Concepts, IoT security overview, Security frameworks for IoT, Privacy in IoT network.
(Text Book- 1, chapter-10)

UNIT – V	**08 Hours**

IOT CASE STUDIES:
Introduction, Design Layers, Design Complexity and Designing Using cloud PaaS, IoT/IIot Applications in the Premises, Supply-Chain and customer monitoring, connected car and its applications and services, IoT Applications for Smart Homes, Cities, environment-monitoring and agriculture, case study: smart city streetlights control and monitoring. ***(Text Book-3, chapter -12)***

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

1. **List** the components, significance, and practical applications of IoT across various domains, including its impact on daily life, industry, and technology.
2. **Determine** effective data handling for seamless communication between smart objects and sensor networks to **design, implement**, and manage IoT architectures, encompassing an understanding of layers, complexities, and cloud-based solutions.
3. **Develop** solutions for troubleshooting communication between smart objects in IoT systems by using protocols, access technologies, and network analytics.
4. Explore machine learning, big data tools, and edge streaming analytics, and **interpret** frameworks to safeguard data while addressing privacy concerns.
5. **Analyse** IoT applications in real-world scenarios and case studies such as smart city streetlight control and monitoring.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	2	2	3				3	3	3		2	3
CO2	2	2	2	2	3				3	3	3		2	3
CO3	2	2	2	2	3				3	3	3		2	3
CO4	2	2	2	2	3				3	3	3		2	3
CO5	2	2	2	3	3				3	3	3		2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXTBOOKS:

1. Buyya, Rajkumar, and Amir Vahid Dastjerdi, eds. Internet of Things: Principles and paradigms. Elsevier, 2016.

2. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, by David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry by CISCO

3. Rajkamal," Internet of Things", Tata McGraw Hill publication

4. Arshdeep Bahga and Vijay Madisetti, Internet of Things - A Hands-On Approach

REFERENCE BOOKS:

1. Hakima Chaouchi "The Internet of Things: Connecting Objects", Wiley publication.

2. 2. Donald Norris "The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and Beagle Bone Black", McGraw Hill publication

E-Resources:

1. https://onlinecourses.nptel.ac.in/noc21_cs63/preview

****

NoSQL DATABASE FUNDAMENTALS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT3509	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the evolution and models of NoSQL databases including key-value, document, column-family, and graph types.
2. **Analyze** consistency, availability, scalability, and query capabilities across various NoSQL models.
3. **Apply** appropriate NoSQL models to real-world domains like e-commerce, finance, healthcare, and social platforms.
4. **Evaluate** limitations, schema migration strategies, and real-world case studies to guide architectural decisions.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, groupwork, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. Use ***Flipped Classes***.

UNIT –I	09 Hours

Unit 1: Introduction to NoSQL and Data Models.

Why NoSQL? The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, (Mostly) Standard Model, Impedance Mismatch, Application and Integration Databases Attack of the Clusters, The Emergence of NoSQL, Key Points, Aggregate Data Models Aggregates Example of Relations and Aggregates, Consequences of Aggregate Orientation, Key-Value and Document Data Models, Column-Family Stores, Summarizing Aggregate-Oriented Databases, Further Reading, Key Points.

TextBook1: 1.1 to 1.6 and 2.1 to 2.6

UNIT – II	07 Hours

Unit 2: Key-Value Databases.

Key-Value Databases, What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preferences, Shopping Cart Data, When Not to Use, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets.

TextBook1: 8.1 to 8.4

UNIT – III	08 Hours

Unit 3: Document-Oriented Databases

Document Databases, What Is a Document Database? Features, Consistency, Transactions, Availability, Query Features, Scaling Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, E-Commerce Applications, When Not to Use, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure.

TextBook1: 9.1 to 9.4

UNIT – IV	**08 Hours**

Unit 4: Column-Family and Graph Databases

Column-Family Stores What Is a Column-Family Data Store? Features Consistency Transactions Availability Query Features Scaling Suitable Use Cases Event Logging Content Management Systems, Blogging Platforms Counters Expiring Usage When Not to Use

Graph Databases What Is a Graph Database Features Consistency Transactions Availability Query Features Scaling Suitable Use Cases Connected Data Routing, Dispatch, and Location-Based Services Recommendation Engines When Not to Use

TextBook1: 10.1 to 10.4 and 11.1 to 11.4

UNIT – V	**07 Hours**

Unit 5: Scaling, Security and Enterprise Applications

Schema Migrations Schema Changes Schema Changes in RDBMS Migrations for Green Field Projects Migrations in Legacy Projects Schema Changes in a NoSQL Data Store Incremental Migration Migrations in Graph Databases Changing Aggregate Structure Further Reading Key Points, Use cases in finance, e-commerce, healthcare, Case studies: Netflix, Facebook, Amazon.

TextBook1: 12.1 to 12.5 and Reference Book1.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Understand** the need and architecture of NoSQL databases and **Analyze** the aggregate data models, **Apply** key-value and document data model concepts to real-world use cases (banking, Healthcare).	L3
2	Understand the structure, features, and constraints of key-value databases. **Analyze** scenarios where key-value stores are best suited or unsuitable. **Apply** key-value modeling to real-world use cases like session and cart data. **Evaluate** limitations of key-value stores in complex transaction or query scenarios.	L4
3	**Understand** the structure and functionality of document databases. **Analyze** document database features like consistency, scalability, and querying. Apply document data modeling to real-world applications like CMS and analytics. **Evaluate** when document databases are unsuitable due to transactional or structural constraints.	L5
4	**Understand** the architecture and basic principles of Column-Family and Graph databases. **Analyze** the consistency, scalability, and querying mechanisms in Column-Family and Graph data models. **Apply** suitable database types to use cases like event logging, analytics, and graph traversal. **Evaluate** limitations of Column-Family and Graph databases in specific use cases.	L5

5	**Understand** the evolution of schema management from RDBMS to NoSQL systems. **Analyze** strategies for schema migration in modern application development. **Apply** NoSQL schema modeling to greenfield and legacy systems. **Evaluate** use cases and real-world case studies for schema migration success/failure. And **Evaluate** enterprise applications and best practices using NoSQL	L5

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	2	-	-	-									1
CO2	2	3	3	2	2									1
CO3	2	3	3	2	3									1
CO4	2	3	3	3	3									1
CO5	2	2	3	2	3									1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. NoSQL Distilled by Pramod Sadalage & Martin Fowler.

REFERENCE BOOKS:

2. Seven Databases in Seven Weeks by Eric Redmond
3. MongoDB University Courses (https://university.mongodb.com/)
4. Neo4j Tutorials and Documentation

E-Resources:

- YouTube: NoSQL vs SQL – Programming with Mosh
- Coursera – Big Data Specialization (UC San Diego)
- YouTube: Redis in 100 Seconds (Fireship)
- Apache Cassandra Docs
- Cassandra MOOC – DataStax Academy
- Amazon DynamoDB Developer Guide
- MongoDB Official Docs
- MongoDB Crash Course – Traversy Media
- MongoDB University Free Courses
- Design Patterns in MongoDB
- YouTube: Intro to Graph Databases – Neo4j
- Neo4j GraphAcademy – Free Courses
- Use Case: LinkedIn's Graph

- Designing Data-Intensive Applications (Martin Kleppmann)
- Case Study: Netflix NoSQL Architecture
- Facebook TAO – Graph Storage
- YouTube: Horizontal Scaling, Sharding Explained
- Coursera – Big Data, NoSQL Courses
- edX – Cloud Computing & NoSQL
- Udemy – MongoDB, Redis, Cassandra, Neo4j
- GeeksforGeeks NoSQL Tutorial

****

DevSecOps FUNDAMENTALS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT3510	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Acquire** the basics of the infrastructure important to utilizing DevOps tools.
2. **Understand** security mechanisms both in the internet and how to authorize users and maintain credentials securely.
3. **Acquire** the controlling access through automation, fault injection test.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

6. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
7. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, groupwork, focused listening, formulating questions, notetaking, annotating, and roleplaying.
8. Show ***Video/animation*** films to explain functioning of various concepts.
9. Encourage ***Collaborative*** (Group Learning) Learning in the class.
10. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
11. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
12. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
13. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT –I: Introduction to DevSecOps	**06 Hours**

Before Devops: The Software Development Life Cycle, What is DevSecOps?, Introducing Automatoonz, Cloud Infrastructure: Secure by Default, Move Fast, Secure Fast: The Importance of Automation, DevSecOps Culture. Setting Up Your Environment

(Text Book-1: Chapter 1: 1.1 to 1.6, Chapter 2)

UNIT – II: Securing Your Infrastructure	**10 Hours**

What makes infrastructure secure?, Preventing unwanted access with IAM permissions, detecting misconfigurations, better than a cure: Implementing preventive controls, Implementation.

Logging and Monitoring : What are Logging and Monitoring and Why do they matter?, attack styles, log types, log storage, detecting anomalies, remediation with AWS config, correlating user activity with cloud trial, **network monitoring with an Amazon VPC**

UNIT – III : Controlling access through automation	**10 Hours**

The Principle of Least Privilege, Fine-Tuning access controls, The IAM pipeline. Fault Injection Test: Distributed Systems, Methods for minimizing downtime, Chaos Engineering,haos Engineering in AWS environments, Chaos Engineering at Automatoonz, AWS Fault Injection Simulator Experiment Examples.

Course Outcomes:

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

1. **Explain** the basics of the infrastructure important to utilizing DevOps tools.
2. **Understand** security mechanisms both in the internet and how to authorize users and maintain credentials securely.
3. **Acquire** the controlling access through automation, fault injection test

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	-	-	-	3	2	-	-	2	2	-		-	3
CO2	2	-	2	-	2	-	-	-	-	3	-		-	-
CO3	2	-	-	-	-	-	-	-	-	3	-		-	2
CO4	2	-	2	-	3	-	-	-	-	3	-		-	-
CO5	3	-	-	-	2	-	-	-	3	3	-		-	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. "Security as code" by BK Sarthak Das, Virginia Chu
2. The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win" by Gene Kim, Kevin Behr, and George Spafford
3. The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations" by Gene Kim, Patrick Debois, John Willis, and Jez Humble A comprehensive guide to implementing DevOps practices, including integrating security into the DevOps process.
4. Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations" by Nicole Forsgren, Jez Humble, and Gene Kim Offers insights into how to achieve high performance in technology organizations through DevOps and Lean principles

E-Resources:

1. https://nptel.ac.in/courses/106/101/106101060/

2. https://www.w3schools.in/definition/what-is-nosql#:~:text=NoSQL%20can%20be%20defined%20as%20an%20approach%20to,models%20having%20flexible%20schemas%20to%20build%20modern%20applications.

E-Resources:

1. **https://www.practical-devsecops.com/wp-content/uploads/2020/05/DevSecOps- Professional-Datasheet-v1.5-detailed.pdf**
2. http://nptel.ac.in/courses
3. https://www.bing.com/videos/riverview/relatedvideo?q=devsecops&mid=3519503E252CDDAFC3473519503E252CDDAFC347&FORM=VIRE

Activity Based Learning (Suggested Activities in Class)

1. Development of project using DevOps tools and group discussion.

2. Collaborative Activity is minor project development with a team of 4 students.

****

AWS WEB SERVICES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25CT3511	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Prerequisites: Fundamentals of Cloud Computing and Cloud Security Services

Course Learning Objectives:

This Course will enable students to:

1. To **understand** fundamental concepts and hands-on knowledge of Cloud Computing using AWS Platform.
2. To **implement** the state-of-the-art AWS Networking, Database, Storage Services in real time projects
3. **Apply** Security and privacy in AWS Cloud across various domains like Storage, Database and applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**5 Hours**

Introduction

What is Cloud Computing? How Does Cloud Computing Work? What is AWS? AWS Free Tier, Compute Services: Amazon EC2, Elastic Load Balancing, Auto Scaling

(Text Book: Chapter 1, Chapter 2)

UNIT – II	**5 Hours**

Networking Services

Amazon VPC, Amazon Route 53

(Text Book: Chapter 4, Chapter 8)

UNIT – III	**5 Hours**

AWS Security Shared Responsibility Model, AWS IAM and KMS ***(Text Book: Chapter 1 , Chapter 6)***	
UNIT-IV	**5 Hours**
AWS Storage Services Amazon S3, Amazon EBS, Amazon EFS, Amazon Glacier, AWS Storage Gateway, Amazon Cloud Front ***(Text Book: Chapter 2 , Chapter 3)***	
UNIT-V	**6 Hours**
AWS Database Services, Application Services Amazon RDS, Amazon DynamoDB, Amazon ElastiCache, Amazon Simple Email Service (Amazon SES), Amazon Simple Notification Service (Amazon SNS), Amazon Simple Queue Service (Amazon SQS), Amazon Simple Workflow Service (Amazon SWF) ***(Text Book: Chapter 5 , Chapter 9)***	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Examine** and practice various AWS services to Configure and deploy, a virtual machine using Amazon EC2.	**L4**
2	**Analyze** different VPC architectures and their use cases in real-world scenarios by Examining the network traffic flow within a VPC and between VPCs.	**L4**
3	**Evaluate** the security and compliance of an AWS environment using AWS KMS for encryption and Assess the impact of key management practices on data security and operational efficiency.	**L5**
4	**Implement** versioning and lifecycle policies to manage data in S3.	**L3**
5	**Analyze** the performance characteristics of different EBS volume types to select the appropriate one for a given workload.	**L6**

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	3	2	3	2	2		2	2			2	
CO2	2	2	3	2	3	2	2		2	2			2	

CO3	2	2	3	2	3	2	2		2	2			2	
CO4	2	2	3	2	3	2	2		2	2			2	
CO5	2	2	3	2	3	2	2		2	2			2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Ben Piper, David Clinton, 'AWS certified solutions architect study guide: associate SAA-C02 Exam (Aws Certified Solutions Architect Official: 22 February 2021

REFERENCE BOOKS:

1. John Paul Mueller, "AWS for Developers For Dummies" , Wiley publications.

2. David Clinton and Ben Piper, "AWS Certified Cloud Practitioner Study Guide: CLF-C01 Exam", Wiley publications.

E-RESOURCES :

1. https://aws.amazon.com/education/awseducate/
2. https://www.simplilearn.com/amazon-web-services-getting-started-rar404-article
3. https://www.mygreatlearning.com/academy/learn-for-free/courses/aws-for-beginners1
4. https://www.edureka.co/free-course/learn-aws-basics-online

Activity Based Learning (Suggested Activities in Class)

1. Building a Server less Application
2. Setting Up a CI/CD Pipeline
3. Hands-On Labs
4. Real time application Projects.

****

SUPPLY CHAIN MANAGEMENT & OPERATIONS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 25CT3601	**Credits**	: 02
Hours / Week	: 02 Hours	**Total Hours**	: 26 Hours
L–T–P–S	: 2–0–0–0		

Course Learning Objectives:

This Course will enable students to:

1. To introduce various concepts related to the supply chain management.
2. To educate the importance of supply chain decision such as design, planning, and operations of a firm.
3. To give the students an understanding of the analytical tools necessary to solve supply chain problems.
4. To showcase the ways by which the companies use different supply chain drivers to improve their performance.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. Lecture method means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.

2. Interactive Teaching: *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.

3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**05 Hours**

INTRODUCTION TO SUPPLY CHAIN MANAGEMENT: Supply chain historical perspective - objectives importance decision phases process view impellers of supply chain financial measures of performance drivers of supply chain performance framework for structuring drivers facilities inventory transportation information sourcing pricing.

(Text Book-1: Chapter 1: 1.1 to 1.6)

UNIT – II	**05 Hours**

DESIGNING DISTRIBUTION NETWORKS AND NETWORK DESIGN:

Role of distribution factors influencing distribution network design, design options for a distribution network, online sales and the distribution network, distribution channels in Indian agricultural, FMCG and commodity sectors, role of network design, factors influencing network design decisions, framework for network design decisions.

(Text Book-1: Chapter 4: 4.1 to 4.7, Chapter 5: 5.1 to 5.3)

UNIT – III	**05 Hours**

DEMAND FORECASTING AND MANAGING UNCERTAINTY:

DEMAND FORECASTING: role, characteristics, components, approach, time series, methods, measures of forecast error, role of it. **MANAGING UNCERTAINTY:** safety inventory and its appropriate level, impact of supply uncertainty, aggregation and replenishment policies. (Text Book-1: Chapter 7: 7.1 to 7.6, 7.9, ***Chapter 12: 12.1,12.4, 12.5,12.7)***

UNIT – IV	**05 Hours**

COORDINATION IN A SUPPLY CHAIN:

Lack of supply chain coordination and the Bullwhip effect, effect on performance, obstacles to coordination, managerial levers, continuous replenishment and vendor- managed inventories, collaborative planning, forecasting and replenishment, Indian experience.

(Text Book-1: Chapter 10: 10.1 to 10.7)

UNIT – V	**06 Hours**

TRANSPORTATION AND SOURCING:

Role of transportation, modes and their performance, design options and their trade, offs Tailored transportation. Sourcing In-house or Outsource 3rd and 4th PLs, Scoring and assessing suppliers. Case studies in SCM

(Text Book-1: Chapter 14: 14.1 to 14.4, 14.6, 14.7, Chapter 15: 15.2)

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3	3											3
CO2	3	3	3											3
CO3	3	3	3											2
CO4	3	3	3											3
CO5	3	3	3											3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Sunil Chopra, Peter Meindl and Kalra D V, Supply Chain Management Strategy, Planning and Operation, Pearson, 5th Edition, 2013.
2. Janat Shah, Supply Chain Management : Text and Cases, Pearson, 1st Edition, 2009.

REFERENCE BOOKS:

1. Donald J Bowersox, David J Closs, Bixby Cooper M, Supply Chain Logistics Management, Tata McGraw Hill, 2nd Edition, 2008.
2. Sople V V, Supply Chain Management Text and Cases, Pearson Education, 2012.
3. Jeremy F Shapiro, Modeling the Supply Chain, Cengage Learning, 2nd Edition, 2007.
4. Coyle, Bardi, Longley, The management of Business Logistics A supply Chain Perspective, Thomson Press, 7th Edition, 2006.

Activity Based Learning (Suggested Activities in Class)

1. Demonstration of solution to a problem through real time examples.

****

INTRODUCTION TO COMPUTER NETWORKS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT3602	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39(Th)+ 26(P)Hours
L–T–P-J	:	3–0–2-0			

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the basic principles of computer networking and how computer network hardware and software operate.
2. **Evaluate** the operation and performance of practical data link protocols using the principles of framing, error detection and correction.
3. **Apply** the principles of network layer design to the analysis and evaluation of routing algorithms, congestion control techniques, internetworking and addressing.
4. **Investigate** the basic transport layer facilities and essentials of transport. protocol
5. **Describe** the working of various application layer protocols.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION

Uses of Computer Networks, Types of Computer Networks ,Network Technology, from Local to Global, Examples of Networks , Network Protocols, Reference Models , Standardization

Text Book (1.1 to 1.7 Pg. nos. -1-74)

UNIT – II	**08 Hours**

THE PHYSICAL LAYER

Guided Transmission Media, Wireless Transmission, Using the Spectrum for Transmission, From Waveforms to Bits, Communication Satellites.

Text Book (2.1- to 2.4, 2.8 Pg. nos. - 90-130, 176 - 184)	
UNIT – III	**08 Hours**
THE DATA LINK LAYER Data Link Layer Design Issues, Error Detection and Correction , Elementary Data Link Protocols , Improving Efficiency. **THE MEDIUM ACCESS CONTROL SUB LAYER** The Channel Allocation Problem, Multiple Access Protocols. **Text Book(3.1 to 3.4 , 4.1 to 4.2.1 Pg. nos.: 202 - 238, 268 - 276)**	
UNIT – IV	**08 Hours**
THE NETWORK LAYER Network Layer Design Issues, Routing Algorithms in a Single Network, The Network Layer in the Internet. **Text Book (5.1, 5.2: 5.2.1 to 5.2.6, 5.7: 5.7.1 to 5.7.3. Pg. nos.: 360 – 384, 441- 470)**	
UNIT – V	**08 Hours**
THE TRANSPORT LAYER The Transport Service, Elements of Transport Protocols, **THE APPLICATION LAYER** DNS — The Domain Name System , Electronic Mail , WWW, Streaming Audio and Video **Text Book (6.1-6.1.3, 6.2 ,7: 7.1.1 to 7.1.5, 7.2:7.2.1, 7.3:7.3.1, 7.4:7.4.1 to 7.4.2 Pg. No 501- 509, 513-536, 613-629, 632-635,650-653, 682-687)**	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	**Understanding** the basic concepts of data communications including the key aspects of networking Network Technology, from Local to Global, Examples of Networks , Network Protocols	L2
2	**Explain** the concepts of Guided Transmission Media, Wireless Transmission, Using the Spectrum for Transmission	L2
3	**Illustrate** the concepts of Design Issues, Error Detection and Correction, Elementary Data Link Protocols, Improving Efficiency.	L3
4	**Explain** the Network Layer Design Issues, Routing Algorithms in a Single Network, The Network Layer in the Internet	L2

5	**Discuss** about the Transport Service, Elements of Transport Protocols, The Domain Name System , Electronic Mail , WWW, Streaming Audio and Video	L2

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	-	-	-	-	-	-	-	-	-		1	-
CO2	3	3	3	3	-	-	-	-	-	-	-		2	-
CO3	3	3	3	-	-	-	-	-	-	-	-		2	-
CO4	3	3	3	-	-	-	-	-	-	-	-		1	-
CO5	3	3	3	-	-	-	-	-	-	-	-		1	-

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

INTRODUCTION TO COMPUTER NETWORK LABORATORY

Total Contact Hours: 26

PART A

1. Implement three nodes point – to – point network with duplex links between them. Set the queue size, vary the bandwidth and find the number of packets dropped.
2. Implement transmission of ping messages/trace route over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.
3. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.
4. Implement simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.
5. Implement and study the performance of GSM on NS2/NS3 (Using MAC layer) or equivalent Environment.
6. Implement and study the performance of CDMA on NS2/NS3 (Using stack called Call net) or equivalent environment.

PART B

Implement the following in Java:

7. Write a program for error detecting code using CRC.
8. Write a program to find the shortest path between vertices using bellman-ford algorithm.
9. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present. Implement the above program using as message queues or FIFOs as IPC channels.
10. Write a program on datagram socket for client/server to display the messages on client side, typed at the server side.

11. Write a program for simple RSA algorithm to encrypt and decrypt the data.
12. Write a program for congestion control using a leaky bucket algorithm.

TEXT BOOKS:

1. Andrew S. Tanenbaum, David.J. Wetherall, —Computer Networks||, Prentice-Hall, 6th Edition, 2022.

REFERENCE BOOKS:

1. Behrouz A. Forouzan,—Data Communications and Networking||, TataMcGraw-Hill,5thEdition, 2012.

2. Chwan-Hwa Wu, Irwin, —Introduction to Computer Networks and Cyber Security||, CRC publications, 2014.

3. Douglas E. Comer, —Internetworking with TCP/IP —, Prentice-Hall, 5thEdition, 2011.

E-Resources:

1. http://computer.howstuffworks.com/computer-networking-channel.htm
2. https://www.geeksforgeeks.org/layers-osi-model/
3. https://www.wikilectures.eu/w/Computer_Network
4. https://technet.microsoft.com/en-us/network/default.aspx

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving using group discussion.
2. Flip class activity

****

LEAN START-UP METHODOLOGY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER –V

Subject Code	:	25CT3603	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	52 Hours
L–T–P–S	:	3–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the opportunities and identify the potential problem to build a solution
2. **Devise** the solution to solve the given problem and build core PoC and business model(0).
3. **Explain** the business model iterations based on the Field experience with customers.
4. **Get the idea** of acquiring / retaining customers as well basic finance aspects.
5. **Describe** and illustrate the current and future path forward of your (potential) start-up.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, by following Idea to PoC to deliver to customer(s) leveraging business model(s) and iterate based on their feedback.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Encourage students to apply the learnings ***to the real world*** with on field experience in helping to improve their understanding.

UNIT – I	**10 Hours**
VISION: Start, Define, Learn & Experiment. ***(Text Book-1: Chapter 1 - 4)***	
UNIT – II	**11 Hours**
STEER: Leap, Test, Measure & Pivot ***(Text Book-1: Chapter 5 - 8)***	
UNIT – III	**10 Hours**

ACCELERATE: Batch, Grow, Adapt, Innovate ***(Text Book-1: Chapter 9 – 12)***	
UNIT – IV	**10 Hours**
BUSINESS MODEL PART I Business Model Definition, 9 Building Blocks, Business Model Canvas, Patterns ***(Text Book-2: Page 14 - 125).***	
UNIT – V	**11 Hours**
BUSINESS MODEL PART II Design – Customer insights, Ideation, Prototyping, Story Telling Strategy – BM environment, Evaluating BM's, Manage Process – Business Model Design Process ***(Text Book-2: Page # 126 - 261***)	
Course Outcomes: At the end of the course the student will be able to: **1. Explain** their understanding of Business model leveraging their field experience. **2. Interpret** and evaluate the Business Model(s). **3. Describe** the Lean Start up Methodology with their practical experience. **4. Identify and explain** the success and failures using Business Model(s). **5. Register Start-Up (Optional)**.	

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3												3
CO2	3	3	2											3
CO3	3	3												3
CO4	3	3	2											3
CO5	2	2												3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Eric Ries "The Lean Startup", ISBN–978-0670921607. https://ia801206.us.archive.org/31/items/TheLeanStartupErickRies/The%20Lean%20Startup%20-%20Erick%20Ries.pdf

2. Alex Osterwalder, Yves Pigneur "Business Model Generation Handbook" ISBN-978-0-470-87641-1
http://alvarestech.com/temp/PDP2011/pdf/Business%20Model%20Generation%20(1).pdf

REFERENCE BOOKS:

1. Steve Blank "The Four Steps to the Epiphany: Successful Strategies for Products That Win", ISBN-13 : 978-0989200509

2. The Entrepreneur's Guide to Customer Development: A "Cheat Sheet" to the Four Steps to the Epiphany Paperback – Import, 6 February 2012, **ISBN-13 :** 978-0982743607

E-Resources:

1. https://ia801206.us.archive.org/31/items/TheLeanStartupErickRies/The%20Lean%20Startup%20-%20Erick%20Ries.pdf

2. http://alvarestech.com/temp/PDP2011/pdf/Business%20Model%20Generation%20(1).pdf

3. https://www.getstoryshots.com/books/the-lean-startup-summary/

4. https://www.strategyzer.com/canvas

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving with on field exposure

2. Build Start-up and test the solution with real customers.

****

SMALL-E BUSINESS LAUNCH

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Experience** with a set of tools and methods for setting up an online service.
2. **Knowledge** of the role of multiple functions in creating a small business.
3. **Ability** to coordinate multiple, interdisciplinary tasks to achieve a common objective.
4. **Emphasis** of specific knowledge from other courses through practice and reflection in an action-oriented setting.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**10 Hours**

INTRODUCTION:

Introduction, Identifying Customer Needs, Opportunity Identification & Planning, Idea Generation and Validation, Entrepreneurial Culture, Digital Business Opportunity

(Text Book-1: Chapter 1: 1.1, 1.4 to 1.9, 1.11 to 1.17, 1.22 to 1.28)

UNIT – II	**10 Hours**

Business Plan & its Components, Business Model and Strategies, Selection of Appropriate Business Model, Business Model Canvas, Feasibility ***(Text Book-1: Chapter 3: 3.1 to 3.18)***	
UNIT – III	**09 Hours**
Lean thinking, Specifications, Wireframe Generation and Validation. ***(Text Book-1: Chapter 5: 5.1 to 5.4, Chapter 5: 5.11 to 5.19)***	
UNIT – IV	**10 Hours**
Service-Market Fit, legal Aspects & Entrepreneurial Law, Logistics of goods delivery, Virtual Trade Shows ***(Text Book-1: Chapter 7: 7.2 to 7.17***)	

Course Outcomes:

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

1. Identify customer needs and pain points.
2. Generate concept and validate idea.
3. Use design thinking to create a wire frame of your online service.
4. Launch your app/website.
5. Register your small business.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	3												3
CO2	3	2	2											2
CO3	2	2	2											2
CO4	3	2	2											3
CO5	3	2												3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Fisher, Steven, Duane, Ja-Nae (2016). Start-up Equation – A Visual Guidebook for Building your Start-up, New Delhi: Tata McGraw-Hill

2. Ulrich, K. T. (2019). Product design and development. Tata McGraw-Hill Education, 7th edition

REFERENCE BOOKS:

1. Holt H., David (2017). Entrepreneurship: New Venture Creation, New Delhi: Pearson Education

2. Zimmerer, W. Thomas and Scarborough, M. Norman and Doug Wilson (2009). Essentials of Entrepreneurship and Small Business Management, 5/e; New Delhi: Prentice Hall India

Activity Based Learning (Suggested Activities in Class)

1. Proposal Handout & Presentation

   a. The Problem

   b. Need

   c. Existing Alternatives

   d. Potential Market

2. Wireframe Sketches, Target Specifications

   a. List Critical Customer Needs

   b. For a., Prepare a List of Target Specifications

   c. Prepare Wireframe

****

PRODUCT ENGINEERING & ENTREPRENEURSHIP

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** and understand customer pain points .
2. **Devise** the best possible solution to build PoC to test in agile way.
3. **Explain** the different roles and responsibilities in collaborative product development
4. **Get the idea** of tools and technologies with automation in building product.
5. **Devise** the business plan and business pitch as potential start-ups.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, by following Idea to PoC cycle leading to development of Business Plan and Pitch to potential investors.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**
INTRODUCTION, PRODUCT DEVELOPMENT PROCESS & ORGANIZATION, PRODUCT PLANNING, OPPORTUNITY IDENTIFICATION , IDENTIFYING CUSTOMER NEEDS ***(Text Book-1: Chapter 1 - 5)***	

UNIT – II	**08 Hours**
PRODUCT SPECIFICATIONS, CONCEPT - GENERATIONS, SELECTION, TESTING, PRODUCT ARCHITECTURE ***(Text Book-1: Chapter 6 - 10)***	

UNIT – III	06 Hours
INDUSTRIAL DESIGN, DESIGN FOR ENVIRONMENT, DESIGN FOR MANUFACTURING & SUPPLY CHAIN, ***(Text Book-1: Chapter 11 - 13)***	
UNIT – IV	**08 Hours**
PROTOTYPING, ROBUST DESIGN, PATENTS & INTELLECTUAL PROPERTY, SERVICE DESIGN ***(Text Book-1: Chapter 14 - 17).***	
UNIT – V	**09Hours**
BUSINESS PLAN, BUSINESS MODEL CANVAS (Below mentioned are reference links) ***(https://bklynlibrary.libguides.com/planning/businessplans*** https://bklynlibrary.libguides.com/planning/businessmodels https://bklynlibrary.libguides.com/planning/pitchdecks)	

Course Outcomes:

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

1. **Explain** the algorithm design techniques and standard Asymptotic notations. Analyze non-recursive and recursive algorithms to obtain worst-case running times of algorithms using asymptotic analysis
2. **Interpret** the brute-force, divide-and-conquer paradigms and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize divide-and-conquer algorithms. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.
3. **Describe** the Decrease and Conquer, Transform and Conquer algorithm design techniques and analyze the performance of these algorithms.
4. **Identify and explain** the greedy technique, dynamic-programming paradigm as to when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms and analyze them.
5. **Describe** the Backtracking, Branch and Bound algorithm design paradigms and explain when an algorithmic design situation calls for it. Recite algorithms that employ these paradigms. **Explain** the limitations of algorithmic power.

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3												3
CO2	3	3	2											3
CO3	3	3												3
CO4	3	3												3
CO5	3	3	2											3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PROJECT DELIVERABLES

1. Proposal Handout & Presentation
    1. The Problem
    2. Need
    3. Existing Alternatives
    4. Potential Market
2. PoC / Prototype, Target Specifications
    1. Core solution
    2. Resources (Environment, Automation, Time to Market)
    3. Build PoC / Prototype
3. Create Business Plan and Model(s) and the Business Pitch
4. Register your business (Optional)

TEXTBOOKS:

1. Ulrich, K. and Eppinger, S.; Product Design and Development | 7th Edition, McGraw Hill; ISBN -13978-9390113231 (https://pdfhost.io/v/jcYs4cRR8_Karl_T_Ulrich_Steven_D_Eppinger_Maria_C_Yang_Product_Design_and_Development_2019_1pdf.pdf)

REFERENCE BOOKS:

1. Christensen, C.M.; The Innovator's Dilemma; Harvard Business Review Press; ISBN: 978-1-63- 369178-0

2. Chesbrough, H. W.; Open Innovation; Harvard Business Review Press; ISBN: 978-1-42-210283-1

E-Resources:

1. https://onlinecourses.swayam2.ac.in/imb19_mg01/preview
2. https://bklynlibrary.libguides.com
3. https://www.meetthedrapers.com/season5video
4. https://sharktank.co.in/episodes/

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving as a team building soft skills as well as getting exposure to product development skills and automation as appropriate apart from Business Plan / Model development and build Pitch like Shark tank / Meet the Drapers
2. Demonstration of PoC & Business Pitch

****

GAMING DESIGN FUNDAMENTALS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT3604	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. To familiarize students with the key concepts and processes involved in game design and development.
2. To equip students with the programming skills necessary to develop basic and advanced game functionalities.
3. To provide an in-depth understanding of multiplayer game development, virtual reality, and game optimization techniques.
4. To encourage ethical considerations and innovative thinking in game design and development.

Teaching-Learning Process (General Instructions)

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Encourage ***Collaborative*** (Group Learning) Learning in the class.
4. Adopt ***experiential Learning*** by following hands-on.
5. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

Games and Video Games: What Is a Game? Conventional Games Versus Video Games, Games for Entertainment, Serious Games. ***(Text Book-1: Chapter 1)***

Designing and Developing Games: An Approach to the Task, Key Components of Video Games, The Structure of a Video Game, Stages of the Design Process, Game Design Team Roles, Game Design Documents, The Anatomy of a Game Designer. ***(Text Book-1: Chapter 2)***

UNIT – II	**08 Hours**

The Major Genres: What Is a Genre? The Classic Game Genres. Understanding Your Player: VandenBerghe 's Five Domains of Play, Demographic Categories, Gamer Dedication, The Dangers of Binary Thinking. ***(Text Book-1: Chapter 3,4)***

Understanding Your Machine: Home Game Consoles, Personal Computers, Portable Devices other Devices. ***(Text Book-1: Chapter 5)***

UNIT – III	**07 Hours**

Game Concepts: Getting an Idea, From Idea to Game Concept ***(Text Book-1: Chapter 7)***

Game Worlds: What Is a Game World? The Purposes of a Game World, The Dimensions of a Game World, Realism.

(Text Book-1: Chapter 8)
Creative and Expressive Play: Self-Defining Play, Creative Play, Other Forms of Expression, Game Modifications. ***(Text Book-1: Chapter 9)***

UNIT – IV	**08 Hours**

Character Development: The Goals of Character Design, The Relationship Between Player and Avatar, Visual Appearances, Character Depth, Audio Design. ***(Text Book-1: Chapter 10)***
Storytelling: Why Put Stories in Games? Key Concepts, The Storytelling Engine, Linear Stories, Nonlinear Stories, Granularity, Mechanisms for Advancing the Plot, Emotional Limits of Interactive Stories, Scripted Conversations and Dialogue Trees, When to Write the Story, Other Considerations. ***(Text Book-1: Chapter 11)***

UNIT – V	**08 Hours**

General Principles of Level Design: What Is Level Design? Key Design Principles, Layouts, Expanding on the Principles of Level Design, The Level Design Process, Pitfalls of Level Design. ***(Text Book-1: Chapter 16)***
Design Issues for Online Gaming: What Are Online Games? Advantages of Online Games, Disadvantages of Online Games, Design Issues, Technical Security, Persistent Worlds, Social Problems. ***(Text Book-1: Chapter 17)***

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

1. **Develop** a comprehensive understanding of the game design and development process, including the essential components, structure, stages, team roles, and documentation involved in creating a video game**.**
2. **Analyze** player preferences and behaviors using VandenBerghe's Five Domains of Play, demographic categories, and levels of gamer dedication while understanding the limitations of binary thinking**.**
3. **Examine** various forms of creative and expressive play, including self-defining play, creative play, other forms of expression, and game modifications**.**
4. **Analyze** the importance of storytelling in games, key storytelling concepts, various narrative structures, plot advancement mechanisms, emotional limits of interactive stories, and best practices for writing game stories**.**
5. **Examine** the characteristics of online games, including their benefits and drawbacks, design challenges, technical security concerns, persistent worlds, and associated social issues.

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	2	2			2				2	2				3
CO2		2				2		2						3
CO3	2		2	2										3
CO4		2	2			2				2				3
CO5	2	2				2	2							3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Fundamentals of Game Design, Third Edition, by Ernest Adams, Released December 2013, Publisher(s): New Riders, ISBN: 9780133435726.

REFERENCE TEXT BOOKS:

1. Designing Games, A Guide to Engineering Experiences by Tynan Sylvester.

2. Game Design Essentials by Briar Lee Mitchell.

Activity Based Learning (Suggested Activities in Class)

1. Develop prototypes of game mechanics or levels using simple tools.
2. Design and build game levels or environments using level editors or game engines.

****

FUNDAMENTALS OF QUANTUM CRYPTOGRAPHY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT3605	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. Understanding the fundamental concepts of quantum computing.
2. Comprehend the concept of qubits and explore the different types of quantum gates and their operations.
3. Explore methods for sharing classical secrets using quantum states.
4. Explore the uncertainty principles framed as a guessing game and their implications for quantum measurements.
5. Be familiar with modern quantum cryptography – beyond quantum key distribution

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**09 Hours**
Quantum Building Blocks ***(Text Book-1: Chapter 1)*** Single Qubit Quantum Systems ***(Text Book-1: Chapter 2)*** Multiple Qubit Quantum Systems ***(Text Book-1: Chapter 3)*** Measurement of Multiple Qubit States ***(Text Book-1: Chapter 4)***	

UNIT – II	**07 Hours**
Introductory Quantum Algorithms ***(Text Book-2: Chapter 6)*** Algorithms with Super Polynomial Speed-up ***(Text Book-2: Chapter 7)*** Algorithms based on Amplitude amplification ***(Text Book-2: Chapter 8)***	

UNIT – III	**08 Hours**
Quantifying Information, Trace Distance, Min-Entropy ***(Text Book-3: Chapter 2)*** Uncertainty Principles as a Guessing Game, Introduction to Privacy Amplification ***(Text Book-3: Chapter 3.2.5)***, Strong Randomness Extractors, Randomness Extraction using Two Universal Hashing (***Text Book-3: Chapter 2.2.1)***,	
UNIT – IV	**08 Hours**
Introduction to Key Distribution ***(Text Book-1: Chapter 2.4)***, Quantum Key Distribution, Public Channel (***Text Book-3: Chapter 3,3.2.4)***, Error Correction and the Cascade Protocols (***Text Book-3: Chapter 4.2)***	
UNIT – V	**07 Hours**
BB84 Protocol (***E-resource 1)***, Warmup: Security Against a Classical Eavesdropper, E91 Protocol ***(E-resource 2)*** Purifying Protocols using Entanglement ***(E-resource 3),*** Quantum Key Distribution: Definition and Concepts, Introduction to Device Independent Quantum Cryptography ***(E-resource 4 & 5)***	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Apply** the fundamental toolbox to **design** and **analyze** quantum protocols	L3
2	**Apply** different algorithms to solve Quantum problems.	L3
3	**Evaluate** the performance and reliability of untrusted quantum devices.	L5
4	**Describe** the principles and mechanisms of quantum key distribution protocols.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	2	2										
CO2	2		2											
CO3	2			2										

CO4	2		2											

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Quantum Computing – A Gentle Introduction- Eleanor Rieffel and Wolfgang Polak, MIT Press, 2011
2. Introduction to Quantum Computing- Philip Kaye, Raymond Laflamme, Michele, Oxford University Press, 2007
3. Applied Quantum Cryptography- Kollmitzer, Christian, Pivk, Mario, Springer-Verlag Berlin Heidelberg, 2010

REFERENCE BOOKS:

1. Quantum Cryptography and Computing, Ryszard Horodecki, Sergei Ya. Kilin, Janusz Kowalik, IOS Press, 2010

E-Resources:

1. https://www.youtube.com/watch?v=STqCrSPBNRU
2. https://journals.sagepub.com/doi/full/10.1177/1550147718778192
3. https://www.sciencedirect.com/science/article/pii/S2095927321007842?via%3Dihub
4. https://www.nature.com/articles/s41467-021-23147-3#Sec1
5. https://ocw.tudelft.nl/course-lectures/7-1-1-device-independent-cryptography/

Activity Based Learning (Suggested Activities in Class)

1. Understanding and Implementing quantum cryptography algorithms in python, playing with key exchange protocols.
2. Demonstration of solution to a problem through programming.

****

EDGE COMPUTING WITH IOT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 25CT3606		**Credits**	:	03
Hours / Week	: 03 Hours		**Total Hours**	:	39 Hours
L–T–P–S	: 2–0–0–2				

Course Learning Objectives:

This Course will enable students to:

1. Understand cloud computing fundamentals
2. Explore edge computing and IoT essentials
3. Develop data analytics skills for real-time applications.
4. Evaluate edge networking and protocol choices.
5. Acquire knowledge of edge computing tools and strategies.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**07 Hours**

Introduction to Cloud Computing: Cloud Computing in a Nutshell, Roots of Cloud Computing, Layers and Types of Clouds, Desired Features of a Cloud, Cloud Infrastructure Management , Infrastructure as a Service Providers, Platform as a Service Providers, Challenges and Risks.
(Reference Book-1: Chapter 1)

Introduction to Computing:

The Major Impacts of Computing, Parallel Computing, Distributed Computing, Cluster Computing, Utility Computing, Cloud Computing, Other Computing Paradigms. ***(Text Book-1: Chapter 1)***

UNIT – II	**08 Hours**

Edge Computing and Its Essentials: Introduction, Edge Computing Architecture, Background Essentials: IoT Devices. ***(Text Book-1: Chapter 2)***
IoT and Edge Computing Definition and use cases: History of the IoT, IoT potential, Definition of the Internet of Things, ***(Text Book-3: chapter 1)***
Cloud and Fog Topologies:
Cloud service model, public, private and hybrid cloud, the OpenStack cloud architecture. Constraints of cloud architecture for IoT. ***(Text Book-3: Chapter 11)***
IoT and Edge Security. ***(Text Book-3: Chapter 13)***

UNIT – III	09 Hours

EDGE ANALYTICS:
Types of Data, Data Analytics, Goals of Data Analytics, Domains Benefiting from Big Data Analytics, Real-Time Applications of Data Analytics, Phases of Data Analytics, Data Collection and Pre-Processing, Machine Learning-Model Building, and Performance Evaluation. ***(Text Book-1: Chapter 3: 3.1 to 3.6).***
Edge platforms: Virtualization, Containers, Use cases for edge computing ***(Text Book-3: Chapter 8)***

UNIT – IV	07 Hours

Edge Routing and Networking:
TCP/IP network functions at the edge, Edge-level network security, Software-defined networking ***(Text Book-3: Chapter 9)***
Edge to Cloud Protocols:
Protocols, MQTT, MQTT-SN, Constrained Application Protocol, Other protocols, Protocol summary and comparison. ***(Text Book-3: Chapter 10)***

UNIT – V	08 Hours

CHALLENGES AND OPPORTUNITIES IN EDGE COMPUTING:
Programmability, Naming, Data Abstraction, Service Management, Privacy and Security, Application, Distribution Scheduling Strategies, Business Model, and Optimization Metrics. ***(Text Book-2: Chapter 5: 5.1 to 5.9***)
Existing Edge Computing Tools What Is Your Role in Edge Computing?, Virtualization, Resource Management, Developing Platforms for Edge Computing. ***(Text Book-2: Chapter 6: 6.1 to 6.4***)

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

1. **Determine** the fundamental concepts of cloud computing, including layers, types of clouds, desired features, and challenges related to managing cloud infrastructure and able to **differentiate** between Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) models.
2. **List** components of edge computing architecture, **use** IoT devices with edge computing, and interpret the historical context and potential impact of IoT.

3. **Develop** data analytics skills by performing data collection, pre-processing, applying machine learning techniques for model building, and recognizing scenarios where real-time data analytics is beneficial.
4. **Evaluate** edge computing networking and protocol choices, TCP/IP network functions at the edge, address edge-level network security concerns, and compare and contrast edge-to-cloud protocols such as MQTT, MQTT-SN, and Constrained Application Protocol (CoAP).
5. **Acquire** knowledge of Edge computing tools and strategies, and engage in discussions about programmability, naming, data abstraction, service management, privacy, security, application distribution, scheduling strategies, business models, and optimization metrics related to edge computing. Additionally, students can identify and explore existing tools, virtualization techniques, and resource management strategies used in edge computing.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	2	2	2								2	1
CO2	3	3	2	2	2								2	1
CO3	3	2	3	2	2								2	1
CO4	2	2	2	3	2								2	1
CO5	2	2	2	2	3								2	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. "EDGE COMPUTING Fundamentals, Advances and Applications"- Anitha Kumari G. Sudha Sadasivam D. Dharani M. Niranjanamurthy CRC Press Taylor and Francis K publication.
2. "Edge Computing: A Primer"- Jie Cao, Quan Zhang , Weisong Shi, Springer Publications.
3. IoT and Edge Computing for Architects, Perry Lea, second edition.

REFERENCE BOOKS:

1. "CLOUD COMPUTING Principles and Paradigms", Rajkumar Buyya, Wiley publications.

E-Resources:

1. https://onlinecourses.nptel.ac.in/noc24_cs66/preview
2. https://www.coursera.org/learn/security-at-the-edge-first-course-1

****

DATA ENGINEERING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT3607	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–SL	:	2–0–0–2			

Prerequisites:

Linear Algebra and Differential Equations, Single and Multi-Variable Calculus, Transforms and Numerical Techniques, Discrete Mathematics and Graph Theory, Probability and Statistics, Data Structures, Proficiency in either Python or R programming language.

Course Learning Objectives:

This Course will enable students to:

1. **Familiarize** with Data Cleaning and Data Processing.
2. **Gain** knowledge of Map Reduce and Data Visualization.
3. **Study** classification models, and clustering methods.
4. **Explore** tools to process the natural language data.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. Use ***Flipped Classes***.

UNIT – I	**13 Hours**

Data Mining: What is Data Mining? , Statistical Modeling, Machine Learning, Computational Approaches to Modeling, Summarization, Feature Extraction, Statistical Limits on Data Mining, Total Information Awareness, Bonferroni's Principle, An Example of Bonferroni's Principle, Exercises for Section, Things Useful to Know, Importance of Words in Documents, Hash Functions, Indexes, Secondary Storage, The Base of Natural Logarithms, Power Laws ***(Book 1, Chapter 1, Pg. No. 1, 1.1, 1.2, 1.3)***

Case Studies: E-Commerce Transactions, Removing Duplicates in Financial Datasets, IoT Data Cleaning for Smart Cities. ***(e resources 4, 5 & 6)***

UNIT – II	**13 Hours**

Map Reduce & the new software stack, Distributed file system, Map Reduce, Algorithms using Map Reduce, Extensions to Map Reduce, The communication cost model, Complexity Theory of Map Reduce. ***(Book 1, Chapter 2, Pg. No. 19, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6)***

Mining Social-Network Graphs: Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs***. (Book 1, Chapter 10, Pg. No. 325, 10.1, 10.2, 10.3, 10.4)***
Case Studies: Sentiment Analysis Using Map-Reduce, Principles of Effective Data Visualization ***(e resources 7 & 8)***

UNIT – III	**13 Hours**

Visualizing Data: matplotlib, Bar charts, Line charts, Scatterplots**,**
(Book 2, Chapter 3, Pg. No. 43)
Clustering: The Idea, The Model, Example: Meetups, Choosing k, Example: Clustering Colors, Bottom-up Hierarchical Clustering ***(Book 2, Chapter 20, Pg. No. 263)***

Natural Language Processing: Word Clouds, n-Gram Language Models, Grammars, An Aside: Gibbs Sampling, Topic Modeling, Word Vectors, Recurrent Neural Networks, Example: Using a Character-Level RNN ***(Book 2, Chapter 21, Pg. No. 279)***
Case Studies: Tracking Website Traffic Over Time, Market Share Distribution.
(e resources 9 & 10)

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Infer** Preprocessing of the data and cleaning the data.	L1
2	**Illustrate** map reduce solutions.	L2
3	**Formulate** ideas to Visualize data and cluster the data for analysis.	L3
4	**Interpret** the Natural Language	L2

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	2	2											2	2
CO2	2	2	2	2										2

CO3	2	2	2	3	3									2
CO4	2	2	3											2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Jure Leskovek, Anand Rajaraman, Jeffrey Ullman, 2014, "Mining of Massive Data Sets", Cambridge University Press

2. Joel Grus, "Data Science from Scratch", O'Reilly, Second Edition

REFERENCE BOOKS:

1. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, EMC Education Services, Wiley

E-Resources:

1. https://www.simplilearn.com/pgp-data-science-certification-bootcamp-program
2. https://wesmckinney.com/book/data-cleaning
3. https://datagy.io/pandas-data-cleaning/
4. https://link.springer.com/chapter/10.1007/978-3-319-92267-6_3
5. https://dev.to/bshadmehr/cleaning-and-preprocessing-financial-data-with-pandas-a-comprehensive-guide-588k
6. https://www.particle.io/iot-guides-and-resources/smart-cities-iot/
7. https://journals.sagepub.com/doi/10.1155/2015/417502
8. https://www.sciencedirect.com/science/article/pii/S2666389920301896
9. https://sitechecker.pro/traffic-checker/
10. https://fastercapital.com/content/Market-share-distribution--How-to-distribute-your-market-share-and-optimize-your-channel-strategy.html

****

Game Development Using Unity

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT3608	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Understand the fundamental concepts of game development and the Unity engine.
2. Write and debug basic C# scripts to control game behavior and interactions.
3. Apply object-oriented programming principles in the context of game development.
4. Create, manipulate, and manage game objects and their components within Unity.
5. Utilize prefabs for efficient game object instantiation and management.
6. Implement realistic physics and collision detection in game projects.
7. Design and manage game scenes and levels effectively.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. Project-Based Learning:
   Engage students in hands-on projects where they create their own games from concept to completion. This approach helps students apply theoretical knowledge to practical scenarios and encourages creative problem-solving.
2. Flipped Classroom:
   Provide instructional videos and reading materials for students to review outside of class, reserving class time for interactive activities such as coding exercises, group discussions, and Q&A sessions. This method maximizes student engagement and allows for more personalized instruction.
3. Collaborative Learning:
   Promote teamwork by assigning group projects where students can collaborate on game development tasks. This helps students develop communication and project management skills and fosters a sense of community within the class.
4. Gamification:
   Incorporate game-like elements into the learning process, such as point systems, leaderboards, and badges for completing milestones. Gamification can increase motivation and make learning more enjoyable.
5. Peer Teaching:
   Encourage students to present their projects and explain their coding strategies to the class. Peer teaching reinforces learning and allows students to gain different perspectives and insights.
6. Industry Interaction:
   Invite guest speakers from the game development industry to share their experiences and insights. Organize workshops and webinars to expose students to real-world practices and trends.
7. Iterative Feedback:
   Provide regular, constructive feedback on student projects and assignments. Encourage iterative improvement by allowing students to refine and resubmit their work based on feedback.
8. Use of Simulation Tools:
   Integrate simulation and visualization tools to demonstrate complex concepts such as physics interactions and AI behaviors. Visual aids can enhance understanding and retention of abstract concepts.
37. Code Reviews and Pair Programming:

Implement code review sessions where students review each other's code to identify errors and suggest improvements. Pair programming can also be used to enhance collaboration and learning efficiency.	
UNIT – I : Introduction to Unity and Basic Concepts	**05Hours**
Introduction to Game Development, Introduction to Unity Engine, History and Evolution, Key Features and Capabilities, Downloading and Installing Unity Hub, Installing Unity Editor, Setting Up the Development Environment, Understanding the Unity Interface, Key Panels: Scene, Game, Hierarchy, Inspector, Project, and Console, Customizing the Layout, Starting a New Project, Project Structure and File Organization, Importing Assets	
UNIT – II Basics of C# Programming and Unity Scripting	**06Hours**
Basic Syntax and Structure, Variables and Data Types, Operators and Expressions, Conditional Statements, Loops: for, while, do-while, Error Handling with Try-Catch, Classes and Objects, Inheritance and Polymorphism, Encapsulation and Abstraction, Using C# Scripts in Unity, The MonoBehaviour Class, Key Methods: Start(), Update()	
UNIT – III Game Objects, Components, and Physics	**05 Hours**
Game Objects Overview, Creating and Manipulating Game Objects, Transform Component, What Are Components?, Adding and Configuring Components, Writing and Attaching Scripts, Creating and Using Prefabs, Instantiating Objects at Runtime, Managing Prefab Instances, Understanding Physics in Unity, Rigidbodies and Colliders, Physics Materials and Joints	
UNIT – IV Scenes, Levels, and Animation	**08 Hours**
Creating and Saving Scenes, Navigating Between Scenes, Lighting and Scene Settings, Level Design Principles, Placing and Arranging Game Objects, Using ProBuilder for Level Design, Introduction to Unity's Animation System, Creating and Editing Animations, Animator Controller and State Machines, Blend Trees, IK (Inverse Kinematics), Animating with Code	
UNIT – V User Interface, Audio, AI, and Final Steps	**15 Hours**
UI Components Overview, Creating UI Elements: Text, Buttons, Images, Canvas and UI Hierarchy, Handling User Input, Creating Menus and HUDs, Scene Transitions and Navigation, Audio Sources and Audio Listeners, Importing and Using Audio Clips, 3D Sound and Spatial Audio, Audio Mixing and Effects, Scripting Audio Behaviors, Introduction to Game AI, Basic AI Concepts: State Machines, Pathfinding, Implementing Simple AI Behaviors, Advanced AI Techniques: Behavior Trees, Flocking, and Steering Behaviors, Profiling Tools, Reducing Memory Usage, Optimizing Scripts and Assets, Building and Packaging Your Game, Cross-Platform Development, Testing and Quality Assurance, Game Distribution Platforms, Marketing Strategies for Indie Games, Post-Release Support and Updates.	

Course Outcomes:

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

1. Exploit the Unity engine's fundamentals for developing the Unity interface, Key Panels, and Layout Customization.
2. Apply object-oriented programming principles in game development.
3. Implement physics and collisions in Unity to create realistic interactions.
4. Develop user interfaces using Unity’s UI system.
5. Implement audio features and effects to enhance the gaming experience.

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3	2	2	3									3
CO2	3	3	2	2	3									3
CO3	3	3	2	2	3									3
CO4	3	3	2	2	3									3
CO5	3	3	2	2	3									3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. "Unity in Action: Multiplatform Game Development in C# with Unity 5" by Joe Hocking
2. "Learning C# by Developing Games with Unity 2020" by Harrison Ferrone

REFERENCE BOOKS:

1. "Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C#" by Jeremy Gibson Bond
2. "Unity Game Development Cookbook: Essentials for Every Game" by Paris Buttfield-Addison, Jon Manning, and Tim Nugent

E-Resources:

6. https://www.coursera.org/specializations/game-design-and-development
7. **https://docs.unity3d.com/Manual/index.html**
8. **https://www.linkedin.com/learning/unity-3d-essential-training**

Activity Based Learning (Suggested Activities in Class)

1. Set up Unity environment, explore the interface, and write basic C# scripts.
2. Develop game mechanics (e.g., scoring systems, game rules) using C# scripts.

****

QISKIT AND QUANTUM CIRCUITS: A HANDS-ON APPROACH

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT3609	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Explain** Qiskit and Qiskit metal, Qiskit Metal Workflow, Quantization Methods Overview.
2. **Explain** Quantum single qubit gates, and introducing quantum circuits, Multi-qubit gates, Quantum entanglement, Bell State.
3. **Demonstrate** End-to-end design for neural network on quantum circuit.
4. **Explain** Quantum Algorithm (finding a secret number, Quantum Fourier transform (QFT), adding number with and without using QFT, Quantum Phase Estimation (QPE), QISKIT tool for dataset.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**13 Hours**

Quantum computer programming:

Introduction to python, Building circuits, CNOT, visualize the Circuit, import and load the data, access backends. Qiskit, installation, testing jupiter notebook with simple python code, Quantum states, qubits, single qubit gates.

UNIT – II	**13 Hours**

Qubits and gates : What happens when gate is applied to qubits? Visualization of Algorithms and gates.	
UNIT – III	**13 Hours**
Deutsch Jozsa Algorithm: Computing mathematical functions on quantum computers, binary numbers and quantum states,arbitrary mathematical functions as unitaries, constant and balanced function on a single bit, Deutsch's algorithm set-up, quantum computing jargon, Deutsch's problem.	

Course Outcomes:

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

1. Building and visualizing circuits, import and loading the data
2. **Explain** Quantum single qubit gates, and introducing quantum circuits, Multi-qubit gates, Quantum entanglement, Bell State.
3. **Demonstrate** mathematical functions on quantum computers, Deutsch's algorithm set-up

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3												3
CO2	3	3	2											3
CO3	3	3												3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Anany Levitin, "Introduction to the Design & Analysis of Algorithms", 2nd Edition, Pearson Education, 2011.
2. Thomas H. Cormen, Charles E.Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, PHI, 2014.
3. Phillip Kaye, Raymond Laflamme et. al., An introduction to Quantum Computing, Oxford University press, 2007. (1) Chris Bernhardt, Quantum Computing for Everyone, The MIT Press,Cambridge, 2020
4. David McMahon-Quantum Computing Explained-Wiley-Interscience , IEEE Computer Society (2008)

REFERENCE BOOKS:

1. Horowitz E., Sahni S., Rajasekaran S, "Computer Algorithms", Galgotia Publications, 2001.
2. R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, "Introduction to the Design and Analysis of

Algorithms A Strategic Approach", Tata McGraw Hill, 2005…….

3. Quantum Computation and Quantum Information, M. A. Nielsen &I.Chuang, Cambridge University Press (2013).
4. Quantum Computing, A Gentle Introduction , Eleanor G. Rieffel and Wolfgang H. Polak MIT press (2014).
5. MichaelA. Nielsen and Issac L. Chuang,"Quantum Computation and Information, Cambridge (2002).
6. Mikio Nakahara and Tetsuo Ohmi,"Quantum Computing", CRC Press (2008). 3. N. David Mermin,"Quantum Computer Science",Cambridge (2007).

E-Resources:

1. https://www.classcentral.com/classroom/youtube-quantum-computer-programming-w-qiskit-45695
2. https://www.cl.cam.ac.uk/teaching/1920/QuantComp/Quantum_Computing_Lecture_7.pdf
3. https://nptel.ac.in/courses/106/101/106101060/
4. http://cse01-iiith.vlabs.ac.in/
5. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
6. https://www.coursera.org/specializations/algorithms
7. https://www.cmlab.csie.ntu.edu.tw/~wcchen/QAnotes/Introduction%20to%20Qiskit%207.pdf

Reference Paper:

1. Zlatko K. Minev, Thomas G. McConkey, Maika Takita, Antonio Corcoles, Jay M. Gambetta, Circuit quantum electrodynamics (cQED) with modular quasi-lumped models. (2021)

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solution to a problem through programming.

****

BUILDING INTELLIGENT IoT SYSTEM

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT3610	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–1			

Course Learning Objectives:

This Course will enable students to:

1. Understand the fundamentals of IoT and edge computing.
2. Set up and configure ThingSpeak channels for data collection.
3. Analyze and visualize IoT data using ThingSpeak.
4. Implement real-time analytics and trigger actions based on data.
5. Explore edge computing architectures and protocols.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION:

Introduction to ThingSpeak the IoT platform with MATLAB analytics.

GET STARTED:

Collect Data in a New Channel, Analyze Your Data, Act on Your Data.

CONFIGURE ACCOUNTS AND CHANNELS:

Channel Configurations, Account Configurations.

UNIT – II	09 Hours

WRITE DATA TO CHANNEL:
Choose How to Write Data, Write Data with Libelium Hardware, Write Distributed Sensor Data from The Things Network, Write Data with Arduino and Particle Photon, Write Data with ESP8266, Write Data with ESP32, Write Data with Raspberry Pi.

READ DATA FROM CHANNEL:
Choose How to Read Data, Read Data with Arduino or Particle Photon, Read Data with ESP8266, Read Data with ESP32.

UNIT – III	07 Hours

PREPARE AND ANALYZE DATA:
Generate MATLAB Code, Prepare Data, Analyze Historical Data, Analyze Live Data.

UNIT – IV	08 Hours

VISUALIZE DATA:
Add Visualizations to Your Channel or Your Website, Visually Compare Data of Different Types, Visually Compare Data of the Same Type, Visualize Time Series Data, Visually Compare Location Data.

UNIT – V	07 Hours

ACT ON DATA: Trigger Actions on your ThingSpeak Data, Use ThingSpeak to Act on Other Websites, Analyze and Display Data, Act on Your Devices with TalkBack.

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

1. **Determine** role of ThingSpeak as an IoT platform and its integration with MATLAB analytics.
2. **Acquire** the skills to collect data in a new ThingSpeak channel, analyze data within ThingSpeak, and implement actions based on data.
3. **Determine** suitable methods for writing data to ThingSpeak channels (using various hardware platforms) and retrieving data from ThingSpeak channels.
4. **Design** MATLAB code for data analysis, prepare data for analytics, and create visualizations within ThingSpeak.
5. **Set up** triggers based on ThingSpeak data, interact with other websites using ThingSpeak, analyze and display data, and implement actions on devices using TalkBack.

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3	2	2	2								2	1
CO2	3	2	3	2	2								2	1
CO3	2	2	2	3	2								2	1
CO4	2	2	2	2	3								2	1
CO5	2	2	2	2	3								2	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

E-Resources:

1. ThingSpeak: Setting up IoT Dashboard (Complete Tutorial) (ucbeginner.com)
2. Learn More - ThingSpeak IoT

Assessment

1. Quizzes and exams
2. Practical assignments (data collection, analysis, and visualization)
3. Final project: Design and implement an IoT application using ThingSpeak and edge computing principles

****

EFFECTIVE DATA TRANSFORMATION WITH DBT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code	:	25CT3611	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	2–0–0–2			
Prerequisites:		Basics of SQL, Data warehousing concepts, Git			

Course Learning Objectives:

This course will enable students to:

1. **Acquire** the basics of the Apache DBT (Data Build Tool).
2. **Acquire** skills required to manage data transformation workflows using DBT.
3. **Acquire** the covers the fundamental concepts, setup, and advanced features of DBT, including data modelling, testing, documentation, and deployment

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I: Introduction to DBT	**08 Hours**
Introduction to DBT and its features, Comparison with traditional ETL tools, Setting up **DBT:** Installation and configuration, Setting up a DBT project. ***(Text Book-1: Chapter 1: 1.1 to 1.4, 1.6, Chapter 2: 2.1,2.2, 2.4)***	
UNIT – II: DBT Basics	**07 Hours**

DBT Basics: DBT Core Concepts**:** Models, sources, seeds, and snapshots, Directory structure and file organization, Building and running models, Writing SQL models, Running and testing models ***(Text Book-1: Chapter 3: 3.1 to 3.3).***
UNIT – III: Data Transformation and Modeling **08 Hours**
What is and why do we need data modeling?. Conceptual, Logical and Physical data models, Entity Relationship Modelling, Modelling Use cases and patterns, Modelling Styles and Architecture, Common Problems in Data Models. ***(Text Book-2 : Chapter 3).***
UNIT – IV: Agile Data Engineering with DBT **07 Hours**
Writing Maintainable code, Working with Dimensional Data, Delivering consistency in data, Delivering Reliability in data ***(Text Book-2 : Chapter 6,7,8,9).***
UNIT – V: Deployment and Operations **09 Hours**
Technical requirement, Designing your deployment automation, Advanced Automation-hooks and run operations, Documentation. ***(Text Book-2 : Chapter 12).***

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
CO1	**Understand** the fundamental concepts and setup of DBT.	L2
CO2	**Develop** and manage basic DBT models and workflows.	L6
CO3	**Apply** data transformation techniques and advanced modeling in DBT.	L3
CO4	**Analyse** Agile Data Engineering with DBT	L4
CO5	**Deploy** DBT projects and manage operational aspects.	L3

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	2	2	3	2	3	2	2	-	2	2	-		2	-
CO2	2	2	3	2	3	2	2	-	2	2	-		2	-
CO3	2	2	3	2	3	2	2	-	2	2	-		2	-
CO4	2	2	3	2	3	2	2	-	2	2	-		2	-
CO5	2	2	3	2	3	2	2	-	2	2	-		2	-

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXTBOOKS:

1. Data Build Tool(DBT) Comprehensive guide for beginners by John Parker .
2. Data Engineering with DBT: A Practical guide to building a cloud based, pragmatic and dependable data platform with SQL by Roberto Zagni.

E-Resources:

3. http://nptel.ac.in/courses
4. https://www.bing.com/videos/riverview/relatedvideo?q=devsecops&mid=3519503E252CDDAFC3473519503E252CDDAFC347&FORM=VIRE

Activity Based Learning (Suggested Activities in Class)

1. Working with Real-World Data Challenges
2. Scenario-Based Problem Solving
3. Project-Based Learning
4. Case Studies

****

MINOR PROJECT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT3612	**Credits**	:	02
Hours / Week	:	04 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	0–0–0–4			

Course Learning Objectives:

This Course will enable students to:

1. To apply theoretical knowledge to solve practical problems.
2. Develop professional project management and communication skills.
3. Able to work in teams and present the project work.

DESCRIPTION:

1. Each B.Tech Project must be carried out by a group of students at the Institute. To ensure uniform participation of each student, the group size should be preferably at least 3 but not more than 4 students.
2. Each project activity must be supervised by the faculty members of the Institute. These faculty members are termed Project Guides.
3. In case the project is of multi-disciplinary nature, the Project group can be formed consisting of students from other Departments. But there must be at least one student and a project Guide from the Department who is offering the Project.
4. The topic proposed by both the guide and the student team should be approved by the department chairman and the department project coordinator to proceed further. A degree of industrial input and involvement will be encouraged and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
5. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
6. The following criteria will be checked by the department chairman to approve for the project proposal:
    a. Department staff as Project guide
        i. Ability to provide direction to the student in the chosen field of interest to formulate a suitable title of the project.
        ii. Ability to design an appropriate strategy and methodology to carry out the Project by the team.
        iii. Ability to provide and evaluate the strong literature review document for the chosen topic.
        iv. Ability to train students on paper / technical writing skills
    b. Student Team
        i. To be dedicated and committed to work on the project by sharpening the existing and learning new technical skills.

ii. To be committed to completing the project and participate in hackathons and project exhibitions.

7. Phase-1 comprises of Literature Survey, Problem identification, Objectives and Methodology.
8. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department based on the rubrics

Course Outcomes:

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

1. Demonstrate practical problem-solving skills by analyzing requirements, designing solutions, implementing algorithms, and testing software systems in real-world scenarios.

2. Develop critical thinking skills, enabling them to evaluate the feasibility, effectiveness, and efficiency of proposed solutions and make informed decisions based on evidence and reasoning.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	3	3	3	3	3	3	3	3	3		3	3
CO2	3	3	3	3	3	3	3	3	3	3	3		3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

PRODUCT ANALYTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25CT4701	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	52 Hours
L–T–P–S	:	3–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Analyse** and understand User Behaviour .
2. Explore the skills to leverage product analytics for data driven decision making.
3. **Get the idea** of tools and technologies around Product Analytics.
4. **Describe** and illustrate the ways to improve Product performance and optimize user experience..

Teaching-Learning Process (General Instructions)

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Encourage ***Collaborative*** (Group Learning) Learning in the class.
4. Adopt ***experiential Learning*** by following hands-on .
5. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

Introduction to Product Analytics

- Understanding the significance of product analytics in modern business.
- Key concepts: metrics, KPIs, and data-driven decision-making.
- Role of Product Analysts and their responsibilities.

Resources:

Article: "A Beginner's Guide to Product Analytics" by Mixpanel
Video: "Introduction to Product Analytics" by Heap

UNIT – II	**08 Hours**

Data Collection and Tools

- Exploring data sources and collection methods.
- Introduction to Google Analytics for tracking user behaviour.
- Ensuring data accuracy, privacy, and compliance.

Resources:

- Google Analytics Academy (Free online courses)
- Article: "Getting Started with Google Analytics" on Analytics Help

UNIT – III	**07 Hours**

Defining Metrics and KPIs

Defining meaningful metrics and KPIs.
Building a measurement framework for effective analysis.

Practical exercise: Identifying KPIs for a social media platform.
Resources: Article: "Choosing the Right Product Metrics" by Amplitude

UNIT – IV	08 Hours

User Behaviour Analysis

- Mapping user journeys and funnel analysis.
- Cohort analysis to measure user retention.
- Hands-on: Analyzing user behavior using cohort analysis.

Resources:
Article: "Cohort Analysis in Google Analytics" by Loves Data
Article: "User Funnel Analysis: A Step-by-Step Guide" by Mixpanel

UNIT – V	08 Hours

A/B Testing and Experimentation, Customer Segmentation, Personalization, Performance

- Understanding A/B testing principles.
- Practical guide to A/B testing with Google Optimize.
- Hands-on: Designing and running an A/B test.

Resources:
Google Optimize (Free tool for A/B testing)
Article: "A/B Testing: A Comprehensive Guide" by CXL

- Segmentation strategies for targeted analysis.
- Leveraging data for personalized user experiences.
- Case studies of successful personalization.

Resources:
Article: "The Power of Personalization in Product Development" by SmarterHQ.
Case Study: "How Netflix Uses Segmentation for Personalization" by Kissmetrics.

- Measuring product success and health.
- Churn analysis and strategies for reducing churn.
- Sentiment analysis of user feedback.

Resources:
Tool: Amplitude for product health measurement.
Article: "Mastering Customer Churn Analysis" by Mixpanel.

Course Outcomes:

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

1. **Describe** what is product analytics and why needed.
2. **Explain** analyzing user behavior.
3. **Describe** the role of product analytics in data driven decision making.
4. **Identify and explain** the business case with example reflecting product performance improvement and optimize user experience

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3												3
CO2	3	3	2											3
CO3	3	3												3
CO4	3	3	2											3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PROJECT / COURSE DELIVERABLES

2. Presentation
    1. The PoC running
    2. Parameters tracked with alternatives
3. PoC / Prototype, Target Specifications
    1. Using Any tool with few parameters
4. Business Case for Product Analytics

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving as a team building soft skills as well as getting exposure to product analytics skills as appropriate.
2. Demonstration of PoC

****

LEAN START-UP METHODOLOGY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the opportunities and identify the potential problem to build a solution
2. **Devise** the solution to solve the given problem and build core PoC and business model(0).
3. **Explain** the business model iterations based on the Field experience with customers.
4. **Get the idea** of acquiring / retaining customers as well basic finance aspects.
5. **Describe** and illustrate the current and future path forward of your (potential) start-up.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, by following Idea to PoC to deliver to customer(s) leveraging business model(s) and iterate based on their feedback.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Encourage students to apply the learnings ***to the real world*** with on field experience in helping to improve their understanding.

UNIT – I	**06 Hours**

VISION:

Start, Define, Learn & Experiment.

(Text Book-1: Chapter 1 - 4)

UNIT – II	**06 Hours**

STEER:

Leap, Test, Measure & Pivot ***(Text Book-1: Chapter 5 - 8)***	
UNIT – III	**07 Hours**
ACCELERATE: Batch, Grow, Adapt, Innovate ***(Text Book-1: Chapter 9 – 12)***	
UNIT – IV	**10 Hours**
BUSINESS MODEL PART I Business Model Definition, 9 Building Blocks, Business Model Canvas, Patterns ***(Text Book-2: Page 14 - 125).***	
UNIT – V	**10 Hours**
BUSINESS MODEL PART II Design – Customer insights, Ideation, Prototyping, Story Telling Strategy – BM environment, Evaluating BM's, Manage Process – Business Model Design Process ***(Text Book-2: Page # 126 - 261***)	

Course Outcomes:

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

1. **Explain** their understanding of Business model leveraging their field experience.
2. **Interpret** and evaluate the Business Model(s).
3. **Describe** the Lean Start up Methodology with their practical experience.
4. **Identify and explain** the success and failures using Business Model(s).
5. **Register Start-Up (Optional)**.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3												3
CO2	3	3	2											3
CO3	3	3												3
CO4	3	3	2											3
CO5	2	2												3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Eric Ries "The Lean Startup", ISBN–978-0670921607.

   https://ia801206.us.archive.org/31/items/TheLeanStartupErickRies/The%20Lean%20Startup%20-%20Erick%20Ries.pdf

2. Alex Osterwalder, Yves Pigneur "Business Model Generation Handbook" ISBN-978-0-470-87641-1 http://alvarestech.com/temp/PDP2011/pdf/Business%20Model%20Generation%20(1).pdf

REFERENCE BOOKS:

1. Steve Blank "The Four Steps to the Epiphany: Successful Strategies for Products That Win", ISBN-13 : 978-0989200509

2. The Entrepreneur's Guide to Customer Development: A "Cheat Sheet" to the Four Steps to the Epiphany Paperback – Import, 6 February 2012, **ISBN-13 :** 978-0982743607

E-Resources:

1. https://ia801206.us.archive.org/31/items/TheLeanStartupErickRies/The%20Lean%20Startup%20-%20Erick%20Ries.pdf
2. http://alvarestech.com/temp/PDP2011/pdf/Business%20Model%20Generation%20(1).pdf
3. https://www.getstoryshots.com/books/the-lean-startup-summary/
4. https://www.strategyzer.com/canvas

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving with on field exposure
2. Build Start-up and test the solution with real customers.

****

PRODUCT ANALYTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyse** and understand User Behaviour .
2. Explore the skills to leverage product analytics for data driven decision making.
3. **Get the idea** of tools and technologies around Product Analytics.
4. **Describe** and illustrate the ways to improve Product performance and optimize user experience..

Teaching-Learning Process (General Instructions)

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Encourage ***Collaborative*** (Group Learning) Learning in the class.
4. Adopt ***experiential Learning*** by following hands-on .
5. Discuss how every ***concept can be applied to the real world*** – and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

Introduction to Product Analytics

- Understanding the significance of product analytics in modern business.
- Key concepts: metrics, KPIs, and data-driven decision-making.
- Role of Product Analysts and their responsibilities.

Resources:

Article: "A Beginner's Guide to Product Analytics" by Mixpanel
Video: "Introduction to Product Analytics" by Heap

UNIT – II	**08 Hours**

Data Collection and Tools

- Exploring data sources and collection methods.
- Introduction to Google Analytics for tracking user behaviour.
- Ensuring data accuracy, privacy, and compliance.

Resources:

- Google Analytics Academy (Free online courses)
- Article: "Getting Started with Google Analytics" on Analytics Help

UNIT – III	**07 Hours**

Defining Metrics and KPIs

Defining meaningful metrics and KPIs.
Building a measurement framework for effective analysis.

Practical exercise: Identifying KPIs for a social media platform.

Resources:

Article: "Choosing the Right Product Metrics" by Amplitude

UNIT – IV	08 Hours

User Behaviour Analysis

- Mapping user journeys and funnel analysis.
- Cohort analysis to measure user retention.
- Hands-on: Analyzing user behavior using cohort analysis.

Resources:

Article: "Cohort Analysis in Google Analytics" by Loves Data

Article: "User Funnel Analysis: A Step-by-Step Guide" by Mixpanel

UNIT – V	08 Hours

A/B Testing and Experimentation, Customer Segmentation, Personalization, Performance

- Understanding A/B testing principles.
- Practical guide to A/B testing with Google Optimize.
- Hands-on: Designing and running an A/B test.

Resources:

Google Optimize (Free tool for A/B testing)

Article: "A/B Testing: A Comprehensive Guide" by CXL

- Segmentation strategies for targeted analysis.
- Leveraging data for personalized user experiences.
- Case studies of successful personalization.

Resources:

Article: "The Power of Personalization in Product Development" by SmarterHQ.

Case Study: "How Netflix Uses Segmentation for Personalization" by Kissmetrics.

- Measuring product success and health.
- Churn analysis and strategies for reducing churn.
- Sentiment analysis of user feedback.

Resources:

Tool: Amplitude for product health measurement.

Article: "Mastering Customer Churn Analysis" by Mixpanel.

Course Outcomes:

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

1. **Describe** what is product analytics and why needed.
2. **Explain** analyzing user behavior.
3. **Describe** the role of product analytics in data driven decision making.
4. **Identify and explain** the business case with example reflecting product performance improvement and optimize user experience

Table: Mapping Levels of Cos to Pos / PSOs														
Cos	**Program Outcomes (Pos)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3												3
CO2	3	3	2											3
CO3	3	3												3
CO4	3	3	2											3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PROJECT / COURSE DELIVERABLES

1. Presentation
    a. The PoC running
    b. Parameters tracked with alternatives
2. PoC / Prototype, Target Specifications
    a. Using Any tool with few parameters
3. Business Case for Product Analytics

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving as a team building soft skills as well as getting exposure to product analytics skills as appropriate.
2. Demonstration of PoC

****

AR/VR & Game-Theoretic

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25CT4702	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Develop professional VR apps using Unity 3D.
2. Run Unity 3D applications in VR on a smartphone.
3. Build tools to help users navigate 3D environments.
4. Learn key usability goals and pitfalls for Virtual Reality.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

MODULE– I	**7 Hours**

INTRODUCTION TO VR:

Definition, Difference between VR & AR, Applications versus Games, Types of VR Experiences, Types of Head-Mounted Displays, Virtual Reality Works? The Benefits Of VR, 3D Computer.

MODULE– II	**8 Hours**

VR & GAME ENGINES:

Overview, Unity3D, VR & 3D Environments, Physics and Gaze Interaction, Virtual World Generator, Human Senses, Perceptual Psychology, Psychophysics, Geometric Modelling, Transforming Rigid Bodies, Game-Theoretic Systems.

MODULE– III	**8 Hours**

3D UI AND LOCOMOTION:

Overview, 3D Menus, Locomotion, Building Interfaces and Locomotion Systems, 3D Rotation Inverses

and Conversions, Look-at & Eye Transforms, Canonical View and Perspective Transforms, Viewport Transforms.
MODULE– IV **8Hours**
3D USER INTERACTION: Overview, 3D Interaction Design, Selection and Manipulation, Building Manipulation Systems, Graphical Rendering, Ray Tracing, Rasterization, Barycentric Coordinates, Anti-Aliasing, Image Warping, Panoramic Rendering.
MODULE– V **8Hours**
WAYFINDING AND VR INPUT: Overview, Implementing Wayfinding, Building Wayfinding Tools, VR Usability and Pitfalls, Velocities, Tracking Systems, Drift Errors, Camera-Feature Detection Model, Perspective point Problem, Brain Machine Interfaces.
Course Outcomes: At the end of the course the student will be able to: 1. Develop Professional VR apps using Unity 3D. 2. Run Unity 3D applications in VR on a smartphone. 3. Create a 3D environment from scratch in game engines.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	2	2	2								3	
CO2	2	2	3	2	2								3	
CO3	2	2	2	3	2								2	
CO4	2	2	2	2	3								2	
CO5	2	2	2	2	3								2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Virtual Reality- Steven M. LaValle, Cambridge University Press
2. Complete Virtual Reality and Augmented Reality Development with Unity- Jesse Glover, Jonathan Lin owes, Packet Publishing (April 17,2019)
3. Understanding Virtual Reality: Interface, Application and Design- William R. Sherman, Alan B. Craig, 1st Edition, September 2002

****

QUANTUM ARTIFICIAL INTELLIGENCE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25CT4703	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Explain fundamental principles of quantum computing.
2. Differentiate between classical and quantum computing paradigms.
3. Apply basic mathematical principles of quantum mechanics.
4. Discuss significance of quantum mechanics in information theory.
5. Implement and analyze basic quantum algorithms.
6. Evaluate advantages of quantum algorithms over classical ones.
7. Develop quantum programs using quantum programming languages.
8. Simulate and test quantum algorithms on quantum hardware.
9. Explore and evaluate applications of quantum computing in AI.
10. Analyze case studies to understand impact of quantum AI in various fields.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

Introduction to Quantum Computing:

Overview of Quantum Computing: Historical background, basic concepts. Differences from classical computing. Quantum Mechanics Essentials: Qubits, superposition, entanglement, quantum gates. Quantum Circuits: Introduction to quantum mechanics and circuits and basic quantum algorithms like Deutsch's algorithm

Textbook 1: *Quantum Computation and Quantum Information* by Michael A. Nielsen, Isaac L. Chuang -

Chapters 1 and 2.

UNIT – II	09 Hours

Quantum Mechanics for AI:
Mathematical Foundations: Linear algebra, Hilbert spaces, quantum states. Quantum Measurement and Operators: Measurement theory, quantum observables, probability amplitudes. Quantum Information Theory: Quantum bits, density matrices, quantum entropy.
Textbook 1: *Quantum Computation and Quantum Information* by Michael A. Nielsen, Isaac L. Chuang - Chapters 2, 3 and 4.
Textbook 2: *Quantum Mechanics for Machine Learning* by Gavin G. Crooks - Chapters 2 and 3.

UNIT – III	07 Hours

Quantum Algorithms:
Basic Quantum Algorithms: Grover's search algorithm, Shor's factorization algorithm. Quantum Machine Learning: Quantum algorithms for machine learning, such as quantum support vector machines, quantum neural networks. Comparison with Classical Algorithms: Efficiency and complexity analysis.
Textbook 1: *Quantum Computation and Quantum Information* by Michael A. Nielsen, Isaac L. Chuang - Chapters 5 and 6.
Textbook 3: *Supervised Learning with Quantum Computers* by Maria Schuld, Francesco Petruccione - Chapters 4 and 5.

UNIT – IV	07 Hours

Quantum Programming and Implementation:
Quantum Programming Languages: Qiskit, Cirq, and other quantum programming frameworks. Developing Quantum Algorithms: Writing and simulating quantum algorithms. Practical Implementation: Building and testing quantum circuits using simulators and actual quantum processors.
Textbook 2: *Quantum Machine Learning: An Applied Approach* by Peter Wittek - Chapters 5 and 6, and **Online Resource**: *Qiskit Documentation* - Relevant sections.

UNIT – V	08 Hours

Quantum AI Applications:
Quantum Computing in AI: Application areas including optimization, search and amplitude amplification, Quantum adiabatic machine learning machine learning, data analysis. Case Studies: Real-world examples of quantum AI applications. Future Trends: Research directions, potential future developments in quantum AI.
Textbook 3: *Supervised Learning with Quantum Computers* by Maria Schuld, Francesco Petruccione - Chapters 6,7 and 8, and **Supplementary Reading**: *Quantum Machine Learning and Optimization in Finance* by Jacob Biamonte, Peter Wittek - Chapters 1,2.

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

1. To **understand** the fundamental concepts of quantum computing and its applications in artificial intelligence.
2. To **explore** the principles of quantum mechanics relevant to quantum computing.

3. To **analyze** various quantum algorithms and their potential advantages over classical algorithms.
4. To **implement** basic quantum machine learning algorithms using quantum programming tools.
5. To critically **evaluate** the current state and future potential of quantum AI technologies.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)													PSOs	
	1	2	3	4	5	6	7	8	9	10	11			1	2
CO1	3		2											2	
CO2	3	3	2											2	
CO3		3	3											2	
CO4			3	3										2	
CO5					3									2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. **Michael A. Nielsen, Isaac L. Chuang**, *Quantum Computation and Quantum Information*, Cambridge University Press, 2010.
2. **Gavin G. Crooks**, *Quantum Mechanics for Machine Learning*, MIT Press, 2019.
3. **Maria Schuld, Francesco Petruccione**, *Supervised Learning with Quantum Computers*, Springer, 2018.

REFERENCES BOOKS:

1. **Peter Wittek**, *Quantum Machine Learning: An Applied Approach*, Academic Press, 2014.
2. **Jacob Biamonte, Peter Wittek**, *Quantum Machine Learning and Optimization in Finance*, Springer, 2020.
3. **Various Authors**, *Ethics and Social Responsibility in AI and Quantum Computing*, [Publisher], [Year].
4. Horowitz E., Sahni S., Rajasekaran S, "Computer Algorithms", Galgotia Publications, 2001.
5. R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, "Introduction to the Design and Analysis of Algorithms A Strategic Approach", Tata McGraw Hill, 2005……..

6. Quantum Computation and Quantum Information, M. A. Nielsen &I.Chuang, Cambridge University Press (2013).

7. Quantum Computing, A Gentle Introduction , Eleanor G. Rieffel and Wolfgang H. Polak MIT press (2014).

8. MichaelA. Nielsen and Issac L. Chuang,"Quantum Computation and Information, Cambridge (2002).

9. Mikio Nakahara and Tetsuo Ohmi,"Quantum Computing", CRC Press (2008). 3. N. David Mermin,"Quantum Computer Science",Cambridge (2007).

ONLINE RESOURCES:

1. *Qiskit Documentation*: https://qiskit.org/documentation/

2. *Quantum Computing Playground*: https://quantum-computing.ibm.com/

3. *Coursera: Quantum Computing for Everyone*: https://www.coursera.org/learn/quantum-computing

4. "Machine Learning with Quantum Computers" Second Edition, Maria Schuld Francesco Petruccione https://doi.org/10.1007/978-3-030-83098-4

5. Qiskit Textbook

6. Xanadu Quantum Codebook

7. Anany Levitin, “Introduction to the Design & Analysis of Algorithms”, 2nd Edition, Pearson Education, 2011.

8. Thomas H. Cormen, Charles E.Leiserson, Ronald L. Rivest, Clifford Stein, “Introduction to Algorithms”, 3rd Edition, PHI, 2014. ……..

9. Phillip Kaye, Raymond Laflamme et. al., An introduction to Quantum Computing, Oxford University press, 2007. (1) Chris Bernhardt, Quantum Computing for Everyone, The MIT Press,Cambridge, 2020

10. David McMahon-Quantum Computing Explained-Wiley-Interscience , IEEE Computer Society (2008)

E-Resources:

1. https://nptel.ac.in/courses/106/101/106101060/
2. http://cse01-iiith.vlabs.ac.in/
3. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
4. https://www.coursera.org/specializations/algorithms

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solution to a problem through programming.

****

CYBER-PHYSICAL SYSTEM

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25CT4704	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Obtain** cyber physical systems fundamentals and principles knowledge as building blocks to promote further design and implementation of more complex real time systems.
2. **Understand** cyber physical systems design for synchronous model with specific case study for arm processor.
3. In what way cyber physical systems are crucial for the optimal performance of asynchronous model.
4. Comprehend the cyber physical systems design and implementation in dynamical models.
5. Hybridization of cyber physical systems which will help the students to anticipate upcoming technologies.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teachingmethods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work,focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with theirown creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION TO CYBER PHYSICAL SYSTEMS:

Introduction to Cyber Physical System, Cyber Physical Systems Design Recommendations ***(Text Book-1: Chapter 1: 1.1 to 1.5)***

Cyber Physical System Requirements, Requirements Engineering and Real time systems, GPU Computing ***(TextBook-1: Chapter 1: 1.9 to 1.15)***,

Wireless sensors network technology, Powerline Communication

(Text Book-1: Chapter 2: 2.1 to 2.6 and 2.8 to 2.13)

UNIT – II	08 Hours

SYNCHRONOUS MODEL:
Reactive Components, Variables, Valuations and Expressions, Execution, Extended-State Machines.
(Text Book-1: Chapter 3: 3.1 to 3.5)
Properties of Components, Final State Components, Combinational Components, Event-Triggered Components
(Text Book-2: Chapter 3: 3.9 to 3.17),
Block diagrams, Input/Output variable renaming, Parallel Composition
(Text Book-1: Chapter 4: 4.1 to 4.8)

UNIT – III	07 Hours

ASYNCHRONOUS MODEL:
Asynchronous model, States, Internal actions, Executions, Deadlocks, Shared Memory,Asynchronous Coordination Protocols, Safety, Safety Specifications, Safety Monitors.

(Text Book-1: Chapter 5: 5.1 to 5.6, Chapter 5: 5.7 to 5.13, Chapter 6: 6.1 to 6.6)

UNIT – IV	08 Hours

DYNAMICAL SYSTEM:
Continuous time model, Composing Components Stability, Linear Systems, Designing Controllers
(Text Book-1: Chapter 8: 8.1 to 8.11).
Open loop versus feedback controller, PID Controller, Analyse techniques
(Text Book-1: Chapter 9: 9.1 to 9.9).

Numerical Solutions, Barrier Certificates. ***(Text Book-2: Chapter 12: 12.1 to 12.8)***,

UNIT – V	08 Hours

HYBRID SYSTEMS
Hybrid dynamic model, Dynamic process ***(Text Book-2: Chapter 1: 1.3 to 1.7***)

Designing hybrid systems, Automated Guide Vehicle, Linear Hybrid automata
(Text Book-2: Chapter 2: 2.1 to 2.5)

Pursuit game, Formal model, Dynamic Automata
(Text Book-2: Chapter 3: 3.4 to 3.9)

Course Outcomes:

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

1. **Understand** the basics of Cyber Physical Systems**.**
2. **Design** Synchronous and Asynchronous model for real time applications**.**
3. **Develop** Deep Understanding on selection of hardware and software's design for designing dynamical systems.
4. **Design and Implement** Cyber Physical Systems and address the problems and limitations for real worldproblems.
5. Come up with cost effective, reliable, robust and feasible designs for real world problems.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3												3
CO2	3	3	3		2									3
CO3	3	3	2		2									2
CO4	3	3	2											3
CO5	3	3												3

3: Substantial (High 2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. Rajeev Alur, "Principles of Cyber Physical systems", 1st Edition, MIT Press, 2015.
2. Raj Rajkumar, "Cyber Physical Systems", 2nd Edition, Elsevier, 2015.

REFERENCE BOOKS:

1. Edward D Lamie, "Computing Fundamentals of Cyber Physical Systems", 2nd Edition, Newnes,
Elsevier.

Activity Based Learning (Suggested Activities in Class)

1. Design Asynchronous problems for real time applications.
2. Demonstration of solution to a problem through real time examples.

****

INTRODUCTION TO DATA VISUALIZATION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25CT4705	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Prerequisites:

Data Structures and Basics of programming language.

Course Learning Objectives:

This Course will enable students to:

1. **To introduce** students with the fundamentals of data visualization and its importance in data analysis and communication.
2. **To provide** students with a practical understanding of different data visualization techniques, tools, and libraries.
3. **Usage of** best practices in data visualization design and development, including basic and advanced design principles.
4. **To expose** students to advanced topics in data visualization, such as interactive visualization, geographic data visualization, and network visualization.
5. **To Develop** data visualization project, from planning and data collection to visualization development and project delivery.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION TO DATA VISUALIZATION:

1.1 Introduction to Data Visualization- What is data visualization? Importance of data visualization, History of data visualization, Types of data visualization

1.2 Visual Perception and Cognition Gestalt Principles of Visual Perception, The Psychology of Color, The Role of Context

1.3 Data Preparation and Design Principles- Data Preparation, Design Principles for Effective Data Visualization

(Text Book-1: Module 1: Chapters 1.1 to 1.3)

UNIT – II	**09 Hours**

DATA VISUALIZATION TECHNIQUES:

2.1 Chart Types and Visualization Techniques- Types of data and their appropriate visual representations, Common Chart Types and their Usage, Custom Visualizations

2.2 Visualization Libraries- Introduction to D3.js, Overview of R ggplot2, Overview of Python Matplotlib

2.3 Dashboarding- Introduction to Dashboards, Design Principles for Effective Dashboards, Tools for Dashboarding: Tableau, PowerBI

(Text Book-1: Module 2: Chapters 2.1 to 2.3)

UNIT – III	**07 Hours**

DATA VISUALIZATION BEST PRACTICES:

3.1 Best Practices in Visualization Design- Common Visualization Mistakes, Making Visualizations Accessible, The Role of Storytelling in Data Visualization

3.2 Effective Visualization Development- Designing for Performance, Scaling Data Visualizations, Using APIs to create Dynamic Visualizations

(Text Book-1: Module 3: 3.1 to 3.2)

UNIT – IV	**08 Hours**

ADVANCED TOPICS IN DATA VISUALIZATION:

4.1 Interactive Visualization- Introduction to Interactive Visualization, Interactive Visualization Techniques, Tools for Interactive Visualization: Bokeh, Plotly

4.2 Geographic Data Visualization- Introduction to Geographic Data Visualization, Mapping Techniques, Tools for Mapping: Leaflet, Mapbox

4.3 Network Visualization- Introduction to Network Visualization, Visualizing Networks, Tools for Network Visualization: Gephi, Cytoscape

(Text Book-1: Module 4: 4.1 to 4.3)

UNIT – V	**07 Hours**

DATA VISUALIZATION PROJECT:

5.1 Project Planning- Identifying a problem statement, Defining Objectives and Deliverables

5.2 Data collection and preparation- Collecting Data, Cleaning and Preparing Data for Visualization

5.3 Data Visualization Development- Visualization Techniques, Designing Effective Visualizations, Developing Interactive Visualizations

5.4 Project Delivery- Critiquing Visualization, Presenting Data Visualization, Final Project Delivery

(Text Book-1: Chapter 5: 5.1 to 5.4)

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Able to **understand** the importance of data visualization in data analysis and communication.	L2

2	**To Apply** basic and advanced design principles to create effective data visualizations for different types of data.	L3
3	**To Implement** different data visualization techniques, tools, and libraries to develop custom visualizations in R and Python.	L3
4	**To organize** interactive visualizations using Bokeh, Plotly, and other tools.	L4
5	**To Design** a data visualization project, including problem identification, data collection and preparation, visualization design and development, and project delivery.	L6

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	3	2	3	2	2	-	2	2	-		2	-
CO2	2	2	3	2	3	2	2	-	2	2	-		2	-
CO3	2	2	3	2	3	2	2	-	2	2	-		2	-
CO4	2	2	3	2	3	2	2	-	2	2	-		2	-
CO5	2	2	3	2	3	2	2	-	2	2	-		2	-

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1."Data Visualization: A Practical Introduction" by Kieran Healy

REFERENCE BOOKS:

1."Storytelling with Data: A Data Visualization Guide for Business Professionals" by Cole Nussbaumer Knaflic

2. "Interactive Data Visualization for the Web: An Introduction to Designing with D3" by Scott Murray.

E-Resources:

1. https://www.coursera.org/learn/data-visualization-communication
2. https://d3js.org
3. https://www.storytellingwithdata.com/blog
4. https://informationisbeautiful.net/

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solution to a problem through programming.

****

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

Unreal Engine Game Development for Beginners

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	: 25CT4706	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–S	: 2–0–0–2		

Course Learning Objectives:

This Course will enable students to:

5. learn the essential principles of programming and build a strong understanding of the value and application of design patterns
6. Produce clean, reusable code using design patterns a series of tools and practices.
7. Design systems with the perfect C++/Blueprint blend for maintainable and scalable systems.
8. Explore a range of different patterns and apply them to project development in Unreal Engine 5.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

9. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
10. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
11. Show ***Video/animation*** films to explain functioning of various concepts.
12. Encourage ***Collaborative*** (Group Learning) Learning in the class.
13. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
14. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
15. Show the ***different ways to design and solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
16. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**07 Hours**
LEARNING FROM UNREAL ENGINE: Supply chain historical perspective - objectives Understanding Unreal Engine 5 and its Layers, Technical requirements, Introducing Unreal Engine 5 Installing Unreal Engine 5 and preparing your development environment, The "Fuzzy" layer – bridging the gap from, C++ to Blueprint, Property Specifiers, Function Specifiers, Useful inheritance, Translating back from Blueprint to C++,Worked example. (Text Book-1: Chapter 1: 1.1 to 1.9)	

UNIT – II	**08 Hours**
HELLO PATTERNS:	
Technical requirements, S.O.L.I.D. principles, Single responsibility, Open-closed, Liskov	

substitution, Interface segregation, Dependency inversion, Exploring solutions to common problems, The moving box problem, The rotating box problem, The cascading cast chain problem, The trade-off , UE5 Patterns in Action – Double Buffer, Flyweight, and Spatial Partitioning (Text Book-1: Chapter 2: 2.1 to 2.7, Chapter 3: 3.1 to 3.5)

UNIT – III	08Hours

PREMADE PATTERNS IN UE5:
Component, Update Method, and Behavior Tree , Understanding and creating components, Applying the update method for prototyping gameplay, working with behavior trees, Creating the AI controller, Creating the Blackboard asset, Building the behavior tree. Anonymous Modular Design : Forgetting Tick ,Event driven systems, case study.
(Text Book-1: Chapter 4: 4.1 to 4.8 Chapter 5 : 5.1 to 5.4)

UNIT – IV	08Hours

CLEAN COMMUNICATION
Interface and Event Observer Patterns- Interfacing communication across classes in UE5, Blueprint interfaces, Interface events versus functions, Interfaces in C++, building an example interface communication, implementing event delegate communication across UE5, Event delegates in Blueprint, Event delegates in C++, Building a useful delegate tool.
(Text Book-1: Chapter 10: 10.1 to 10.7)

UNIT – V	08Hours

USER INTERFACE
UMG Overview, Root Widget, Canvas Panel, Horizontal Box and Vertical Box, Grid Panel and Uniform Grid Panel, Common Widget Properties, Visual Designer, Text Wedge, Button Wedge, Border Widget and Image Widget, Progress Bar Widget, Check Box Widget, Scale Box and Size Box, Creating the HUD, Damage Tint and Collect Item Tint.

Course Outcomes:

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

1. Build a strong understanding of the value and application of design patterns .
2. **Apply** and Produce clean, reusable code using design patterns a series of tools and practices
3. Identify and **analyze** factors that influence decisions in network design**.**
4. Design systems with the perfect C++/Blueprint blend for maintainable and scalable systems.
5. **Examine** case studies and examples of different patterns and apply them to project development in Unreal Engine 5.

Table: Mapping Levels of COs to POs / PSOs		
COs	**Program Outcomes (POs)**	**PSOs**

	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	3		2				3	2	2		3	
CO2	3	3	3		2				3	2	2		3	
CO3	3	3	3		2				3	2	2		2	
CO4	3	3	3		2				3	3	3		3	
CO5	3	3	3		2				3	3	3		3	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Unreal Engine - The Complete Beginner's Course, O"REILLY publishing pvt. ltd
2. Unreal Engine 5 for Beginners: Dive into the world of game development with Unreal Engine 5 to build amazing 3D games By Sargey Rose,

REFERENCE BOOKS:

1. UE5: Fundamentals Vol.1 - Essential Beginner's Guide to Getting Started with Unreal® Engine 5
2. Build your first 3d game: learn collision detection in unreal engine :student guide ONLINE RESOURCE

Activity Based Learning (Suggested Activities in Class)

1. Demonstration of solution to a problem through real time examples.

****

QUANTUM COMPUTING WITH PENNYLANE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25CT4707	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P-S	:	2–0–0-2			

Prerequisites:

1. Basic Python programming skills
2. Linear algebra.
3. Calculus.
4. Quantum mechanics.

Course Learning Objectives:

This Course will enable students to:

1. Comprehend the basic concepts of quantum mechanics and quantum computing.
2. Understand the operational principles of quantum gates and circuits.
3. Setup and use PennyLane for simulating quantum circuits.
4. Explore the basics of quantum machine learning using PennyLane.
5. Implement key quantum algorithms using PennyLane.
6. Apply quantum Fourier transform and Grover's algorithm in problem-solving.
7. Analyze and implement quantum machine learning models.
8. Utilize PennyLane for creating quantum neural networks.
9. Apply quantum computing techniques to solve complex real-world problems.
10. Evaluate the effectiveness of quantum algorithms in practical applications.
11. Understand the current limitations and future potential of quantum computing.
12. Discuss the ethical implications of quantum technology.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different type of teaching methods may be adopted to develop the course outcomes.
2. ***Interactive Teaching***: Adopt the Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make Critical thinking, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's

possible, it helps improve the students' understanding.	
UNIT- I	**07Hrs**
Introduction to Quantum Computing: Quantum Mechanics Fundamentals: Qubits, superposition, and entanglement. Quantum Gates and Circuits: Basic quantum gates, quantum circuits. Quantum Measurement: Quantum state measurement, quantum probability. Quantum vs Classical Computing: Differences and advantages. **Textbook 1**: *Quantum Computing for Computer Scientists* by Noson S. Yanofsky and Mirco A. Mannucci - Chapter 5. **Textbook 2**: *Quantum Computation and Quantum Information* by Michael A. Nielsen and Isaac L. Chuang - Chapter 1.	
UNIT- II	**07Hrs**
Getting Started with PennyLane: Introduction to PennyLane: Overview, installation, and setup. Quantum Circuit Simulation: Creating and simulating quantum circuits. Quantum Machine Learning: Introduction to quantum machine learning with PennyLane. Quantum Computing Ecosystem: Integration with other tools and platforms (e.g., Qiskit, Cirq). **Textbook 3**: *PennyLane Documentation and Tutorials* - Available online. **Supplementary Reading**: *Learning Quantum Computing with Python and Qiskit* by Frank Harkins, Hassi Norlén - Chapter 3.	
UNIT- III	**07Hrs**
Implementing Quantum Algorithms with PennyLane: Basic Quantum Algorithms: Quantum teleportation, superdense coding. Quantum Fourier Transform: Implementing QFT using PennyLane. Quantum Search Algorithms: Grover's algorithm, amplitude amplification. Quantum Error Correction: Basics of error correction, quantum codes. **Textbook 3**: *PennyLane Documentation and Tutorials* - Available online. **Textbook 4**: *Quantum Computing: An Applied Approach* by Jack D. Hidary - Chapter 5,7. **Supplementary Reading**: *Quantum Computing for Everyone* by Chris Bernhardt -Chapter 6.	
UNIT- IV	**07Hrs**
Quantum Machine Learning: Introduction to Quantum Machine Learning: Concepts, applications. Quantum Neural Networks: Basics, creating QNN with PennyLane. Variational Quantum Algorithms: VQE, QAOA. Case Studies: Quantum machine learning applications in real-world scenarios. **Textbook 3**: *PennyLane Documentation and Tutorials* - Available online. **Textbook 5**: *Quantum Machine Learning* by Peter Wittek - Chapter 3. **Supplementary Reading**: *Quantum Computing and Quantum Machine Learning* by Sandor Imre, Ferenc Balazs - Chapter 4.	
UNIT- V	**06Hrs**

Real-World Applications of Quantum Computing: Quantum Computing in Optimization: Portfolio optimization, traffic flow. Quantum Chemistry Simulations: Modeling molecules, chemical reactions. Quantum Cryptography: Quantum key distribution, quantum-secure communication. Case Studies: Real-world applications and industry use cases. **Textbook 5**: *Quantum Machine Learning* by Peter Wittek **Supplementary Reading**: *Quantum Chemistry and Quantum Computations* by E. Ludena - Chapter 8. And **Textbook 6**: *Quantum Computing: A Gentle Introduction* by Eleanor Rieffel, Wolfgang Polak - Chapter 7.

UNIT- VI	**05Hrs**
Challenges and Future Directions in Quantum Computing: Current Challenges: Scalability, error rates, qubit coherence. Future Directions: Quantum supremacy, hybrid quantum-classical systems. Ethical and Societal Implications: Impact on security, privacy, and society. Research Trends: Latest advancements and future research areas. **Textbook 1**: *Quantum Computing for Computer Scientists* by Noson S. Yanofsky and Mirco A. Mannucci - Chapter 10. **Supplementary Reading**: *Introduction to Quantum Technologies* by Julio Benitez - Chapter 5.	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	To **understand** the fundamental principles of quantum computing and quantum mechanics	L2
2	To **explore** the capabilities of PennyLane for **simulating** quantum circuits	L2
3	To **implement** quantum algorithms using PennyLane	L3
4	To **analyze** the integration of quantum computing with machine learning	L3
5	To **apply** quantum computing techniques to solve real-world problems	L3
6	To **evaluate** the challenges and future directions of quantum computing	L3

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	**PSOs**
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**		**1**	**2**
CO1	3	3											2	
CO2		2	2										2	
CO3			3	3	2								2	
CO4				1	3								2	
CO5					2		2	3					2	
CO6	3					3	3				3		2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. **Noson S. Yanofsky, Mirco A. Mannucci**, *Quantum Computing for Computer Scientists*, Cambridge University Press, 2008.
2. **Michael A. Nielsen, Isaac L. Chuang**, *Quantum Computation and Quantum Information*, Cambridge University Press, 2010.
3. **PennyLane Documentation and Tutorials** - Available online.
4. **Jack D. Hidary**, *Quantum Computing: An Applied Approach*, Springer, 2019.
5. **Peter Wittek**, *Quantum Machine Learning*, Academic Press, 2014.
6. **Eleanor Rieffel, Wolfgang Polak**, *Quantum Computing: A Gentle Introduction*, MIT Press, 2011.

SUPPLEMENTARY REFERENCES:

1. **Frank Harkins, Hassi Norlén**, *Learning Quantum Computing with Python and Qiskit*, Packt Publishing, 2020.
2. **Chris Bernhardt**, *Quantum Computing for Everyone*, MIT Press, 2019.
3. **Sandor Imre, Ferenc Balazs**, *Quantum Computing and Quantum Machine Learning*, Springer, 2021.
4. **E. Ludena**, *Quantum Chemistry and Quantum Computations*, CRC Press, 2020.
5. **Julio Benitez**, *Introduction to Quantum Technologies*, Springer, 2021.

****

REAL-TIME CYBER-PHYSICAL SYSTEMS DESIGN USING MATLAB

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25CT4708	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

11. To know about fundamentals of MATLAB tool.
12. To provide an overview to program curve fitting & solve Linear and Nonlinear Equations.
13. To understand the concept and importance of Fourier transforms.
14. To gain knowledge about MATLAB Simulink & solve engineering problems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. Project-Based Learning: Engage students in hands-on projects where they create their own games from concept to completion. This approach helps students apply theoretical knowledge to practical scenarios and encourages creative problem-solving.
2. Flipped Classroom: Provide instructional videos and reading materials for students to review outside of class, reserving class time for interactive activities such as coding exercises, group discussions, and Q&A sessions. This method maximizes student engagement and allows for more personalized instruction.
3. Collaborative Learning: Promote teamwork by assigning group projects where students can collaborate on game development tasks. This helps students develop communication and project management skills and fosters a sense of community within the class.
4. Gamification: Incorporate game-like elements into the learning process, such as point systems, leaderboards, and badges for completing milestones. Gamification can increase motivation and make learning more enjoyable.
5. Peer Teaching: Encourage students to present their projects and explain their coding strategies to the class. Peer teaching reinforces learning and allows students to gain different perspectives and insights.
6. Industry Interaction: Invite guest speakers from the game development industry to share their experiences and insights. Organize workshops and webinars to expose students to real-world practices and trends.
7. Iterative Feedback: Provide regular, constructive feedback on student projects and assignments. Encourage iterative improvement by allowing students to refine and resubmit their work based on feedback.
8. Use of Simulation Tools: Integrate simulation and visualization tools to demonstrate complex concepts such as physics interactions and AI behaviors. Visual aids can enhance understanding and retention of abstract concepts.
9. Code Reviews and Pair Programming: Implement code review sessions where students review each other's code to identify errors and suggest improvements. Pair programming can also be used to enhance collaboration and learning efficiency.

	39 Hours

Introduction to MATLAB Programming: Basics of MATLAB Programming, array operations in MATLAB, loops and execution of control, working with files: Scripts and functions, plotting and programming output, examples.

Numerical Methods and their applications: Curve Fitting: Straight line fit, Polynomial fit.

Numerical Integration and Differentiation: Trapezoidal method, Simpson method.

Linear and Nonlinear Equations: Eigen values, Eigen vectors, Solution of linear algebraic equations using Gauss Elimination and LU decomposition, Solution of nonlinear equation in single variable using Gauss-Siedal and Newton-Raphson method.

Ordinary Differential Equations: Introduction to ODE's, Euler's method, second order RungaKutta method,

MATLAB ode45 algorithm in single variable and multivariables. Transforms: Discrete Fourier Transforms,

Application of MATLAB to analyse problems in basic engineering mechanics, mechanical vibrations, control system, statistics and dynamics of different circuits. MATLAB Simulink: Introduction to MATLAB Simulink, Simulink libraries, development of basic models in Simscape Power Systems

Course Outcomes:

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

1. Able to implement loops, branching, control instruction and functions in MATLAB programming environment.
2. Able to program curve fitting, numerical differentiation and integration, solution of linear equations in MATLAB and solve electrical engineering problems.
3. Able to understand implementation of ODE using ode 45 and execute Solutions of nonlinear equations and DFT in MATLAB.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	2	2	3									3
CO2	3	3	2	2	3									3
CO3	3	3	2	2	3									3
CO4	3	3	2	2	3									3
CO5	3	3	2	2	3									3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Agam Kumar Tyagi, "MATLAB and Simulink for Engineers", OXFORD Higher
2. Dr. Shailendra Jain, "Modeling& Simulation using MATLAB – Simulink", Wiley – India.

REFERENCE BOOKS:

1. Won Y.Tang, Wemun Cao, Tae-Sang Ching and John Morris, "Applied Numerical Methods Using MATLAB", A John Wiley & Sons.
2. Steven T. Karris, "Introduction to Simulink with Engineering Applications", Orchard Publications.

****

TABLEAU FOR DATA ANALYSTS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25CT4709	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Gain proficiency in utilizing Tableau as powerful tools for data visualization and analysis.
2. Acquire the ability to clean, preprocess, and transform raw data into a format suitable for visualization.
3. Develop a variety of visualizations using both basic and advanced techniques.
4. Understand how data visualization is applied in real-world engineering scenarios and various industries.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**09 Hours**
Understanding the basics, visualizing data, Connecting to data in Tableau. ***(Text Book-1: Chapter 1,2)***	

UNIT – II	**09 Hours**
Moving beyond basic visualizations, starting an adventure with calculation and parameters, leveraging level of detail calculations, diving deep with table calculations. ***(Text Book-1: Chapter 3,4,5,6)***	

UNIT – III	**10 Hours**
Telling a data story with dashboards, visual analytics: Trends, clustering, Distributions and Forecasting. ***(Text Book-1: Chapter 8,9)***	

UNIT – IV	**11 Hours**
Advanced Visualization Techniques, Dynamic Dashboards, exploring mapping and advanced geospatial features.	

(Text Book-1: Chapter 10, 11,12)
UNIT – V
Understanding the tableau data model, joins, and blends, structuring messy data to work well in Tableau, Taming data with Tableau Prep. ***(Text Book-1: Chapter 14,15,16)***

Course Outcomes:

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

1. **Apply** the fundamentals of Tableau and visualize data using various charts and dashboards**.**
2. **Interpret** level of detail calculations effectively, and delve deeply into table calculations for comprehensive data analysis in Tableau.
3. **Create** effective data visualizations using dashboards and advanced visual analytics techniques such as trend analysis, clustering, distribution analysis, and forecastin**g.**
4. **Analyze** advanced visualization techniques, geographic features, and mapping to create a dynamic dashboard for in-depth data analysis and presentation using Tableau.
5. **Apply** the Tableau data model, perform joins and blends to organize complex data for appropriate Tableau analysis, and Tableau Prep for data cleaning and transformation.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	2	2	2	2	2				2	2				2
CO2	2	2	2	2	2				2	2				2
CO3	2	2	2	2	2				2	2				2
CO4	2	2	2	2	2				2	2				2
CO5	2	2	2	2	2				2	2				2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Learning Tableau 2022: Create effective data visualizations, build interactive visual analytics, and improve your data storytelling capabilities 5th edition, by Joshua N Milligan.

REFERENCE BOOKS:

1. Mastering Tableau 2023 - Fourth Edition: Implement advanced business intelligence techniques, analytics, and machine learning models with Tableau 4th edition, by Marleen Meier.
2. Tableau For Dummies, 2nd Edition (For Dummies (Computer/tech)) 2nd edition,by Jack A. Hyman.

Activity Based Learning (Suggested Activities in Class)

1. Design and build a dashboard using multiple visualizations.
2. Analyze case studies involving complex data challenges.

****

CAPSTONE PROJECT-PHASE I

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25CT4710	**Credits**	:	05
Hours / Week	:	03 Hours	**Total Hours**	:	65 Hours
L–T–P–S	:	0–0–0–10			

Course Learning Objectives:

This Course will enable students to:

1. To apply theoretical knowledge to solve practical problems.
2. Enhance technical, analytical, and problem-solving skills.
3. Foster teamwork and collaboration skills.
4. Develop creativity and innovation in addressing engineering challenges.
5. To analyze and design the solution to the selected problem statement.

DESCRIPTION:

1. Each B. Tech Project must be carried out by a group of students at the Institute. To ensure uniform participation of each student, the group size should be preferably at least 3 but not more than 4 students.
2. Each project activity must be supervised by the faculty members of the Institute. These faculty members are termed Project Guides.
3. In case the project is of multi-disciplinary nature, the Project group can be formed consisting of students from other departments. But there must be at least one student and a project Guide from the department who is offering the Project.
4. The topic proposed by both the guide and the student team should be approved by the department chairman and the department project coordinator to proceed further. A degree of industrial input and involvement will be encouraged and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
5. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
6. The following criteria will be checked by the department chairman to approve for the project proposal:
   a. Department staff as Project guide
      i. Ability to provide direction to the student in the chosen field of interest to formulate a suitable title of the project.
      ii. Ability to design an appropriate strategy and methodology to carry out the Project by the team.
      iii. Ability to provide and evaluate the strong literature review document for the chosen topic
      iv. Ability to train students on paper / technical writing skills
   b. Student Team
      i. To be dedicated and committed to work on the project by sharpening the existing and

learning new technical skills.

ii. To be committed to completing the project and participate in hackathons and project exhibitions.

8. Phase-1 comprises of Literature Survey, Problem identification, Objectives and Methodology.
9. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the department based on the rubrics

Course Outcomes:

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

1. Demonstrate the ability to apply engineering principles to solve real-world problems.
2. Develop innovative thinking and thereby preparing students for Capstone project

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	3	3	3	3	3	3	3	3	3		3	3
CO2	3	3	3	3	3	3	3	3	3	3	3		3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

CAPSTONE PROJECT-PHASE II

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VIII

Subject Code	:	25CT4801	**Credits**	:	12
Hours / Week	:	03 Hours	**Total Hours**	:	156 Hours
L–T–P–S	:	0–0–0–24			

Course Learning Objectives:

This Course will enable students to:

1. Detailed design of the solution to the problem statement and project management using software engineering skills.
2. Write efficient code and test the code to find any bugs and resolve the same leading to completion and deployment of the project using modern tools.
3. Analyze and synthesize the project results.
4. Demonstrate knowledge and understanding of writing the publication/report.
5. Able to work in teams and present the project work.

DESCRIPTION:

1. The problem statement selected in Major Project Phase-I (VII semester) will be carried in the VIII semester.
2. Phase-0II comprises of the detailed design, implementation, and testing results during the internal and external review.
3. Each Project team needs to submit the technical paper or patent or participate in hackathons and project exhibitions as well as apply for various state and national funding agencies within the stipulated time frame by the university.
4. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department.
5. Additionally, there will be a Semester end evaluation of the work done that would include an internal Faculty and an external academic expert.

Course Outcomes:

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

1. Conduct a survey of several available literature in the preferred field of study to find the recent advances and gaps
2. Implement the mathematics concept and engineering fundamentals, and specialization to design a solution using modern tools for the defined problem.
3. Experimenting and evaluating the results from test data to provide a conclusion to the project work.

4. Demonstrate an ability to work in teams and to prepare quality documents of project work & exhibit technical presentation skills.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	3	3	3	3	3	3	3	3	3		3	3
CO2	3	3	3	3	3	3	3	3	3	3	3		3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

INTERNSHIP

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VIII

Subject Code	:	25CT4802	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	0–0–6–0			

Course Learning Objectives:

This Course will enable students to:

1. To expose students to the industrial environment.
2. To create competent professionals for the industry.
3. To provide possible opportunities to learn, understand and sharpen the real time technical /managerial skills required at the job.
4. To work on a problem assigned by a mentor at industry, prepare action plan and complete within time limit.
5. To learn, create/prepare report for Project/research as used in industry with productive and efficient way .
6. To strengthen industry-institute linkage and increase employability of the students.

Guideline for Internship:

The course includes 16 weeks of on-job training on current industry-relevant problem through supervised self-learning approach The internship is an individual activity. The student should obtain approval from the chairman/supervisor to pursue. A student shall submit a brief proposal about the work to be carried out in the internship, to a coordinator within 3 weeks, after starting the internship.

A comprehensive report is required to be prepared and submit to the department at the end of the semester. A certificate shall be attached with this report duly signed by the competent authority of the industry for the successful completion of the internship. An attendance report shall also be attached with this report. The CIA evaluation will be done by faculty mentor or Industry Supervisor. There is no SEE Exam for this course.

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Understand the modern tools used in the field of Computer science and engineering for product development.	L2
CO2	Demonstrate ethical conduct and professional accountability while working in a team for the benefit of society.	L2
CO3	Understand the resources requirement and planning to facilitate the project success.	L3
CO4	Develop and refine oral and written communication skills.	L3
CO5	Demonstrate knowledge of the industry in which the internship is done.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	3	3	3	3	3	3	3	3	3	3		3	3
CO2	3	3	3	3	3	3	3	3	3	3	3		3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****