



**Nagarjuna College of Engineering &  
Technology, Bengaluru**

An Autonomous Institute, Affiliated to VTU Belagavi

2024 Batch  
Scheme & Syllabus  
of  
III Semester

As per the NEP 2020 Guidelines,  
Choice-Based Credit System  
&  
Outcome-Based Education

**CSE (Data Science)**

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

### **Vision:**

To build a strong technical environment and foster leadership and problem-solving abilities in the domain of Data Science, creating professionals capable of addressing social and technical challenges.

### **Mission:**

1. To equip and expose students with the latest tools and technologies.
2. To instill critical problem-solving capabilities, leadership qualities, research capabilities and to prepare them for global challenges.
3. To establish state-of-the-art laboratories and foster collaborations with leading industries in the field of Data Science.

### **PROGRAM OUTCOMES (POs):** Graduates of the Computer Science and Engineering – Data Science

Program will be able to achieve the following

**PO1: Engineering Knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in Washington Accord Knowledge 1 (WK1) to Washington Accord Knowledge 4 (WK4) respectively to develop the solution of complex engineering problems.

**PO2: Problem Analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

**PO3: Design/Development of Solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

**PO4: Conduct Investigations of Complex Problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8)

**PO5: Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

**PO6: The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

**PO7: Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

**PO8: Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

**PO9: Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

**PO10: Project Management and Finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

**PO11: Life-Long Learning:** Recognize the need for, and have the preparation and ability for

- i) Independent and life-long learning.
- ii) Adaptability to new and emerging technologies and
- iii) Critical thinking in the broadest context of technological change. (WK8)

### **Program Specific Outcome (PSO)**

**PSO1:** Analyze complex computing problems and apply to derive appropriate solutions.

**PSO2:** Design, implement, and evaluate database-oriented, computing-based solutions that address a broad range of requirements in the field of Data Science.

**PSO3:** Communicate and work effectively within diverse teams and professional environments.

### **Program Educational Objectives (PEOs)**

**PEO1:** To work as Data Scientist with an ability to solve wide range of computational problems.

**PEO2:** To work effectively in a diverse and multi-disciplinary field, as a team member or leader to solve the societal problems.

**PEO3:** Engage in self-directed and lifelong learning, continuously updating their skills by adapting emerging techniques, advancing in research and higher studies.

# Nagarjuna College of Engineering and Technology, Bangalore

## B.E. in CSE (Data Science)

### Scheme of Teaching and Examinations 2023

Outcome Based Education (OBE) and Choice Based Credit System (CBCS)  
(Effective from the academic year 2025-26)

III SEMESTER													
Sl. No	Course	Course Code	Course Title	Teaching Department (TD) and Question Paper Setting Board (PSB)	Teaching Hours /Week				Examination				Credits
					Theory Lecture	Tutorial	Practical /Drawing	Self-Study	Duration in hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	S					
1	BSC/PCC	24MATS31	Mathematics for Computer Science	Basic Science	2	2	0		03	50	50	100	3
2	IPCC	24CDI32	Data Structures and Applications	TD:CD PSB:CD	3	0	2		03	50	50	100	4
3	IPCC	24CDI33	Database Management System	TD:CD PSB:CD	3	0	2		03	50	50	100	4
4	PCC	24CDT34	Object Oriented Programming using C++	TD:CD PSB:CD	3	0	0		03	50	50	100	3
5	PCCL	24CDL35	C++ Lab	TD:CD PSB:CD	0	0	2		03	50	50	100	1
6	ESC	24CDT36x	ESC/ETC/PLC	TD:CD PSB:CD	3	0	0		03	50	50	100	3
7	UHV	24UHV37	Social Connect and Responsibility	Any Department	0	0	2		01	100	---	100	1
8	AEC/SEC	24CDL38x	Ability Enhancement Course/Skill Enhancement Course - III		If offered as Theory courses				01	50	50	100	1
					1	0	0						
					If offered as lab. courses				02				
9	MC	24NS39	National Service Scheme (NSS)	NSS coordinator	0	0	2			100	---	100	0
		24PE39	Physical Education (PE) (Sports and Athletics)	Physical Education Director									
		24YO39	Yoga	Yoga Teacher									
<b>Total</b>									<b>21</b>	<b>550</b>	<b>350</b>	<b>900</b>	<b>20</b>
<p><b>PCC:</b> Professional Core Course, <b>PCCL:</b> Professional Core Course laboratory, <b>UHV:</b> Universal Human Value Course, <b>MC:</b> Mandatory Course (Non-credit), <b>AEC:</b> Ability Enhancement Course, <b>SEC:</b> Skill Enhancement Course, <b>L:</b> Lecture, <b>T:</b> Tutorial, <b>P:</b> Practical <b>S=SDA:</b> Skill Development Activity, <b>CIE:</b> Continuous Internal Evaluation, <b>SEE:</b> Semester End Evaluation. K : This letter in the course code indicates common to all the stream of engineering. <b>ESC:</b> Engineering Science Course, <b>ETC:</b> Emerging Technology Course, <b>PLC:</b> Programming Language Course</p>													
<b>Engineering Science Course (ESC/ETC/PLC)</b>													
24CDT361	IT Infrastructure and Management			24CDT363	Supply Chain Management								
24CDT362	Business Process Fundamentals			24CDT364	Human Computer Interaction								
<b>Ability Enhancement Course – III (All are Laboratory Courses 0-0-2)</b>													
24CDL381	Introduction to Android Programming			24CDL383	Introduction to Office Tools								
24CDL382	Advanced Python Programming			24CDL384	Introduction to Linux/Unix Shell Programming								
<p><b>Professional Core Course (IPCC):</b> Refers to Professional Core Course Theory Integrated with practicals of the same course. Credit for IPCC can be 04 and its Teaching– Learning hours (L : T : P) can be considered as (3 : 0 : 2) or (2 : 2 : 2). The theory part of the IPCC shall be evaluated both by CIE and SEE. The practical part shall be evaluated by only CIE (no SEE). However, questions from the practical part of IPCC shall be included in the SEE question paper. For more details, the regulation governing the Degree of Bachelor of Engineering /Technology (B.E./B.Tech.) 2022-23 may please be referred.</p>													
<p><b>National Service Scheme /Physical Education/Yoga:</b> All students have to register for any one of the courses namely National Service Scheme (NSS), Physical Education(PE)(Sports and Athletics), and Yoga(YOG) with the concerned coordinator of the course during the first week of III semesters. Activities shall be carried out between III semester to the VI semester (for 4 semesters). Successful completion of the registered course and requisite CIE score is mandatory for the award of the degree. The events shall be appropriately scheduled by the colleges and the same shall be reflected in the calendar prepared for the NSS, PE, and Yoga activities. These courses Shall not be considered for vertical progression as well as for the calculation of SGPA and CGPA, but completion of the course is mandatory for the award of degree.</p>													

## MATHEMATICS FOR COMPUTER SCIENCE

<b>Course Code</b>	24MATS31	<b>CIE Marks</b>	<b>50</b>
<b>Course Type</b>	Theory	<b>SEE Marks</b>	<b>50</b>
<b>Teaching Hours/Week (L: T: P: S)</b>	2:2:0:0	<b>Total Marks</b>	<b>100</b>
<b>Total Hours of Pedagogy</b>	40 hours	<b>Exam Hours</b>	<b>03</b>
		<b>Credits</b>	<b>03</b>

### Course objectives:

The goal of the course **Mathematics for Computer Science** is to,

1. Introduce the concept of random variables, probability distributions, specific discrete and continuous distributions with practical application in Computer Science Engineering and social life situations.
2. Provide the principles of statistical inferences and the basics of hypothesis testing with emphasis on some commonly encountered hypotheses.
3. Determine whether an input has a statistically significant effect on the system's response through ANOVA testing.

### Teaching-Learning Process (General Instructions)

**These are sample Strategies, which teacher can use to accelerate the attainment of the various course outcomes.**

1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the delivered lessons shall develop students theoretical and applied mathematical skills.
2. State the need for Mathematics with Engineering Studies and Provide real-life examples.
3. Support and guide the students for self-study.
4. You will also be responsible for assigning homework, grading assignments and quizzes, and documenting students' progress.
5. Encourage the students for group learning to improve their creative and analytical skills.
6. Show short related video lectures in the following ways:
  - As an introduction to new topics (pre-lecture activity).
  - As a revision of topics (post-lecture activity).
  - As additional examples (post-lecture activity).
  - As an additional material of challenging topics (pre-and post-lecture activity).
  - As a model solution of some exercises (post-lecture activity).

### Module – I

#### Probability Distributions:

Review of basic probability theory. Random variables (discrete and continuous), probability mass and density functions. Mathematical expectation, mean and variance. Binomial, Poisson and normal distributions- problems (derivations for mean and standard deviation for Binomial and Poisson distributions only).

**08 Hours**

[Text 1: 26.1, 26.2, 26.7, 26.8, 26.9, 26.10, 26.13, 26.14, 26.15, 26.16]

[RBT Levels: L1, L2 and L3]

**Self-Study:** Exponential distribution.

**Applications:** Used for Modeling and prediction, analyzing data, algorithm design, cryptography, error detection, machine learning, computer vision, computer graphics, random number generation and natural language processing.

### Module - II

**Joint probability distribution & Markov Chain:**

**Joint probability distribution:** Joint Probability distribution for two discrete random variables, expectation, covariance and correlation.

**Markov Chain:** Introduction to Stochastic Process, Probability Vectors, Stochastic matrices, Regular stochastic matrices, Markov chains, Higher transition probabilities, Stationary distribution of Regular Markov chains and absorbing states. **08 Hours**

[Text 3: 8.1, 8.2, 8.3, 8.4, 8.5, 8.6, 5.6, 5.7]

[RBT Levels: L1, L2 and L3]

**Self-Study:** Joint Probability distribution for two continuous random variables.

**Applications:** Joint distribution for system design and maintenance decisions. Markov chain for algorithmic design and networking.

**Module – III****Statistical Inference 1:**

Introduction, sampling distribution, standard error, testing of hypothesis, levels of significance, test of significances, confidence limits, simple sampling of attributes, test of significance for large samples, comparison of large samples. Sampling variables, central limit theorem and confidence limit for unknown mean. Test of Significance for means of two large samples. **08 Hours**

[Text 1: 27.1, 27.2, 27.3, 27.4 27.5, 27.6, 27.7, 27.8, 27.9, 27.10, 27.11, 27.12]

[RBT Levels: L1, L2 and L3]

**Applications:** Decision making and problem solving, software testing and quality control

**Module – IV****Statistical Inference 2:**

Sampling of variables-small samples, students ‘t’ distribution, Chi-square distribution as a test of goodness of fit. F-Distribution. **08 Hours**

[Text 1: 27.13, 27.14, 27.15, 27.16, 27.17, 27.18, 27.19 ]

[RBT Levels: L1, L2 and L3]

**Self-Study: Fisher’s Z-Distribution.**

**Applications:** Algorithm performance evaluation, Software testing, Hardware testing, Quality assurance, Biometric systems, Network security, database management, Biomedical informatics, Information retrieval, signal processing and image processing.

**Module – V****Design of Experiments and ANOVA:**

Principles of experimentation in design, Analysis of completely randomized design, randomized block design. The ANOVA Technique, Basic Principle of ANOVA, One-way ANOVA, Two-way ANOVA, Latin-square Design, and Analysis of Co-Variance. **08 Hours**

[Text 2: ]

[RBT Levels: L1, L2 and L3]

**Applications:** Algorithm Optimization, Network performance, Database management, User experience design and Hardware design.

**Teaching-Learning Process for all modules**

**Chalk and Talk/PowerPoint presentation/YouTube videos.**

**Course Outcomes(Course Skill Set):**

**After successfully completing the course, the students will be able to:**

CO1: Understand the basic concepts of probability, random variables, probability distribution and apply suitable probability distribution models for the given scenario.

CO2: Learn the concept of joint distribution and make use of the notion of a discrete-time Markov chain and n-step transition probabilities to solve the engineering application problem.

CO3: Use statistical methodology and tools in the sampling analysis.

CO4: Compute the confidence intervals for the mean of the population by using different tests.

CO5: Apply the ANOVA test related to engineering problems.

**Evaluation Details:**

<b>Theory Component</b>	IAT-1 after completion 45 to 50% Syllabus	25 Marks
	IAT-2 after completion 95 to 100% Syllabus	25 Marks
	Average of two IATs	<b>25 Marks</b>
	CCE-1	25 Marks
	CCE-2	25 Marks
	Average of two CCEs	<b>25 Marks</b>
<b>Grand Total of IAT Marks (min marks 20 / 50)</b>		<b>50 Marks</b>
SEE conducted for 100 and scaled down to 50 (min marks 18/50)		<b>50 Marks</b>
IAT + SEE (min marks 40)		<b>100 Marks</b>

**Suggested Learning Resources:**

**Text Books:**

1. **B. S. Grewal:** "Higher Engineering Mathematics", Khanna publishers, 44th Ed.2021.
2. **Ronald E. Walpole, Raymond H Myers, Sharon L Myers & Keying Ye** "Probability & Statistics for Engineers & Scientists", Pearson Education, 9<sup>th</sup> edition, 2017.
3. **Seymour Lipschutz and Marc Lars Lipson:** "Probability", (Chapters: 5 and 8), McGraw Hill Education (India) Private Limited, Chennai, Special Indian Edition, 2010.

**Reference Books:**

1. **Erwin Kreyszig**, "Advanced Engineering Mathematics", John Wiley & Sons, 9<sup>th</sup> Edition, 2006.
2. **Peter Bruce, Andrew Bruce & Peter Gedeck** "Practical Statistics for DataScientists" O'Reilly Media, Inc., 2<sup>nd</sup> edition **2020**.
3. **G Haribaskaran** "Probability, Queuing Theory & Reliability Engineering", Laxmi Publication, Latest Edition, 2006.
4. **Irwin Miller & Marylees Miller**, John E. Freund's "Mathematical Statistics with Applications" Pearson. Dorling Kindersley Pvt. Ltd. India, 8<sup>th</sup> edition, 2014.
5. **S C Gupta and V K Kapoor**, "Fundamentals of Mathematical Statistics", S Chand and Company, Latest edition.
6. **Robert V. Hogg, Joseph W. McKean & Allen T. Craig.** "Introduction to Mathematical Statistics", Pearson Education 7<sup>th</sup> edition, 2013.
7. **Jim Pitman.** Probability, Springer-Verlag, 1993.
8. **Sheldon M. Ross**, "Introduction to Probability Models" 11<sup>th</sup> edition. Elsevier, 2014.
9. **A. M. Yaglom and I. M. Yaglom**, "Probability and Information". D. Reidel Publishing Company. Distributed by Hindustan Publishing Corporation (India) Delhi, 1983.
10. **P. G. Hoel, S. C. Port and C. J. Stone**, "Introduction to Probability Theory", UniversalBook Stall, (Reprint), 2003.
11. **S. Ross**, "A First Course in Probability", Pearson Education India, 6<sup>th</sup> Ed., 2002.
12. **W. Feller**, "An Introduction to Probability Theory and its Applications", Vol. 1, Wiley, 3<sup>rd</sup> Ed., 1968.
13. **N.P. Bali and Manish Goyal**, A Textbook of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
14. **Veerarajan T**, Engineering Mathematics (for semester III), Tata McGraw-Hill, New Delhi, 2010.

**E-Resources:**

- <http://.ac.in/courses.php?disciplineID=111>
- [http://www.class-central.com/subject/math\(MOOCs\)](http://www.class-central.com/subject/math(MOOCs))
- <http://academicearth.org/>
- VTU e-Shikshana Program
- VTU EDUSAT Program

**Activity-Based Learning (Suggested Activities in Class)/ Practical Based learning**

- Quizzes
- Assignments
- Seminars

## DATA STRUCTURES AND APPLICATIONS

<b>Course Code</b>	<b>24CDI32</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours/Week (L:T:P: S)</b>	<b>3:0:2:0</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>40 hours Theory + 13 Lab slots</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>04</b>	<b>Exam Hours</b>	<b>03</b>

### Prerequisite:

- Basic knowledge of common programming concepts, including loops, arrays, stacks, and recursion.
- Basic knowledge of mathematics, including proof by induction and contradiction.

### Course Objectives:

This course will enable students to:

- Understand the basics of pointers and dynamic memory allocation.
- Learn concepts of structure and its application in programming.
- Gain knowledge to choose the specific linked lists for implementing real world problems.
- Grasp the use of stacks and queues in solving complex problems.
- Acquire knowledge of non-linear data structure like trees.

### Teaching-Learning Process (General Instructions)

These are sample Strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic 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

**Introduction:** Data Structures, Classifications (Primitive & Non Primitive), Data Structures operations, Functions, Review of Pointers and Dynamic Memory Allocation, Arrays, Structures and Unions, Case study on Dynamic allocated arrays.

**Stacks:** Stack, Array Representation of Stacks along with Stack Operations, Stack Applications: Reverse Polish notation, Infix to Postfix conversion, Evaluation of postfix expression, Stacks using Dynamic Arrays.

**08 Hours**

### Module – II

<p><b>Recursion:</b> Introduction to Recursion, Examples of Recursion: Factorial, Fibonacci Sequence, Tower of Hanoi.</p> <p><b>Queues:</b> Queues, ADT- Queue, Array Representation, Queue Operations, Circular Queues, Programming Examples.</p> <p><b>Priority Queues:</b> Single and double ended Priority Queues. <span style="float: right;"><b>08 Hours</b></span></p>	
<b>Module – III</b>	
<p><b>Linked Lists:</b> Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Stacks and Queues implementation using Linked Lists, Doubly Linked lists, Programming Examples. <span style="float: right;"><b>08 Hours</b></span></p>	
<b>Module – IV</b>	
<p><b>Circular lists:</b> Circular Singly and Doubly linked lists; Basic operations: Insert, Delete and Display with programming examples. Application: Adding Polynomials.</p> <p><b>Trees:</b> Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, Postorder, Preorder; <span style="float: right;"><b>08 Hours</b></span></p>	
<b>Module – V</b>	
<p><b>Trees:</b> Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application: Evaluation of Expression.</p> <p><b>Graphs:</b> The Graph Abstract Data Types, Elementary Graph Operations.</p> <p><b>Hashing:</b> Introduction, Static Hashing, Dynamic Hashing <span style="float: right;"><b>08 Hours</b></span></p>	
<b>Teaching-Learning Process for all modules</b>	Chalk and board, Active Learning, PPT Based presentation, Video
<b>LIST OF EXPERIMENTS</b>	
<b>1</b>	Design, develop and execute a program in C based on the following requirements: An EMPLOYEE structure is to contain the following members: Employee_Number (an integer), Employee_Name (a string of characters), Basic_Salary (an integer), All_Allowances (an integer), IT (an integer), Net_Salary (an integer). Write a function to read the data of an employee, to calculate Net_Salary and to print the values of all the structure members. (All_Allowances = 123% of Basic, Income Tax (IT) = 30% of the gross salary (Gross salary = Basic_Salary + All_Allowance), Net_Salary = Basic_Salary + All_Allowances –IT). Display the above data for at least 5 employees.
<b>2</b>	Write a program to Store Roll number of N students. Perform Insert and delete Roll_No at a given valid position (POS) using pointers. Display the status of array elements at any given point of time. Support the program with functions for each operation.
<b>3</b>	Develop an array implementation on stack and perform Push and Pop operations. Check for overflow and underflow conditions. Demonstrate stack implementation to check palindrome. Display the status of the stack for all the operations performed. Support the program with appropriate functions for each of the above operations.
<b>4</b>	Write a Program in C to evaluate the Suffix (Postfix) expression with single digit operands and operators.
<b>5</b>	Implement Circular QUEUE program in C for rainbow colors (VIBGYOR) and perform Insert and Delete operations. Check for overflow and underflow conditions. Display the status of the Circular QUEUE for all the operations performed. Use pointers and functions.
<b>6</b>	Implement a Menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: <i>USN, Name, Avg_Marks</i> <ol style="list-style-type: none"> <li>a. Create N number of Students Data by inserting at end of the list.</li> <li>b. Insert and Delete at front of the list</li> </ol>

	c. Delete at the end of list d. Display the status of SLL d. Demonstration stack and queue e. Exit
<b>7</b>	Design and Develop following operations on Doubly Linked List (DLL) of Employee Data with the fields: <i>SSN, Name, Dept etc.</i> a. Create a Node of N Employees Data by inserting in front. b. Insert a new node to the right of key value. c. Perform Insertion and Deletion at End of DLL d. Display the status of DLL and count the number of nodes e. Exit
<b>8</b>	Design and Develop a program in C for the following operations on Binary Search Tree (BST) of Integers. a. Create a BST of N Integers b. Traverse the BST using Inorder, Preorder and Post Order techniques c. Search a KEY element in BST and display the appropriate message

### Course Outcomes

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

CO1: Define and explain the concepts of Data Structures

CO2: Apply the knowledge of Data Structures and its applications in solving real time problems

CO3: Analyze and identify the output for the given code snippet.

CO4: Develop different methods of organizing the data using either linear or non-linear data structures.

CO5: Study independently as a member of the team and present effectively on the applications of data structures.

### Assessment Details (both IAT and SEE)

<b>Theory Component</b>	IAT-1 after completion 45 to 50% Syllabus	25 Marks
	IAT-2 after completion 95 to 100% Syllabus	25 Marks
	Average of two IATs	<b>25 Marks</b>
	<b>Total 25 Marks : Reduced to 15 Marks</b>	
	CCE-1	25 Marks
	CCE-2	25 Marks
	Average of two CCEs	<b>25 Marks</b>
<b>Total 25 Marks : Reduced to 10 Marks</b>		
<b>Lab Component</b>	Lab Record and execution of programs	<b>15 Marks</b>
	<b>Lab Test at the end of 15<sup>th</sup> week</b>	<b>10 Marks</b>
	<b>Total</b>	<b>25 Marks</b>
<b>Grand Total of IAT Marks</b>		<b>50 Marks</b>
Obtaining 40% of marks in both theory and lab component is essential for appearing for SEE		

### Suggested Learning Resources:

#### Text Books:

- Ellis Horowitz and Sartaj Sahani, Fundamentals of Data Structures in C, 2<sup>nd</sup> Ed, Universities Press, 2014. (Chapter 1: 1.1, 1.2; Chapter 2: 2.2; Chapter 3: 3.2 & 3.4; Chapter 4: 4.3 & 4.8 ; Chapter 5: 5.1 – 5.3, 5.7).
- Reema Thareja, Data Structures using C, 3<sup>rd</sup> Ed, Oxford press, 2012, (Chapter 2: 2.1 – 2.7; Chapter 3: 3.1 – 3.8, Chapter 5: 5.1 to 5.10 & 5.14 to 5.16; Chapter 7: 7.1 to 7.5; Chapter 8: 8.5 – 8.8; Chapter 9: 9.1 to 9.3, 9.7 & 9.8, 9.11 - 9.13).

**Reference Books:**

1. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2<sup>nd</sup> Ed, Cengage Learning, 2014.
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1<sup>st</sup> Ed, McGraw Hill, 2014. Chapters 5.1 – 5.8, 6.6 – 6.7, 8.3.

**Weblinks and Video Lectures (e-Resources):**

1. <https://www.cs.princeton.edu/>
2. <https://www.opendatastructures.org/ods-cpp>
3. <https://www.lib.mdp.ac.in/ebook/DSa>
4. <https://ww.cs-fundamentals.com/data-structures/introduction-to-datastructures.php>
5. <https://www.cprogramming.com/algorithms-and-data-structures.html>
6. <https://online-learning.harvard.edu/course/data-structures-and-algorithms>

## DATABASE MANAGEMENT SYSTEM

<b>Course Code</b>	<b>24CDI33</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours/Week (L:T:P: S)</b>	<b>3:0:2:0</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>40 hours Theory + 13 Lab slots</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>04</b>	<b>Exam Hours</b>	<b>03</b>

### Course Objectives :

#### The Student will be able to:

1. Understand the basics of database concepts, technology, and practice.
2. Gain Knowledge of the relational database design principles.
3. Practice SQL programming through a variety of database.
4. Learn the use of concurrency and transactions in database.
5. Gain knowledge of building database applications for real world problems.

### Teaching-Learning Process (General Instructions)

These are sample Strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic 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

**Introduction to Databases:** Introduction, Characteristics of database approach, Advantages of using the DBMS approach.

**Overview of Database Languages and Architectures:** Data Models, Schemas, and Instances. Three schema architecture and data independence, database languages, and interfaces, The Database System environment.

**Conceptual Data Modelling using Entities and Relationships:** Entity types, Entity sets and structural constraints, Weak entity types, ER diagrams.

**08 Hours**

### Module – II

**Relational Model:** Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations.

**SQL:** SQL data definition and data types, Schema change statements in SQL, specifying constraints in SQL, retrieval queries in SQL, INSERT, DELETE, and UPDATE statements in SQL, Additional features of SQL.

**08 Hours**

### Module – III

<p><b>SQL: Advanced Queries:</b> More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Implementation of different types of Joins in SQL, Views in SQL.</p> <p><b>Functional Dependencies and Normalization:</b> Informal design guidelines for relation schemas, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form.</p> <p style="text-align: right;"><b>08 Hours</b></p>	
<b>Module – IV</b>	
<p><b>Transaction Processing-I:</b> Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions.</p> <p><b>Transaction Processing-II:</b> Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL.</p> <p style="text-align: right;"><b>08 Hours</b></p>	
<b>Module – V</b>	
<p><b>NOSQL Databases:</b> Introduction to NoSQL - CAP Theorem – Data Models - Key-Value Databases - Document Databases- Column Family Stores – Graph Databases – Working of NoSQL Using MONGODB/CASSANDRA.</p> <p style="text-align: right;"><b>08 Hours</b></p>	
<b>Teaching-Learning Process for all modules</b>	Chalk and board, Active Learning, PPT Based presentation, Video
<b>LIST OF EXPERIMENTS</b>	
<b>1</b>	<p>The storage of digital data is increasing day by day. Every big/small organization has started storing their employee details such as name, salary, address, and department under which they are working in their own database. Design a company database that stores the details of Departments, Projects, Employees, and their Dependents in a particular organization using the following relational schema:</p> <p>Relational Schema</p> <p>Employee (ssn, name, salary, sex, super_ssn, address, dno)  Department (dname, dnumber, mgr_ssn)  Dept_Loc (dnumber, dloc)  Project (pname, pnumber, plocation, dnum)  Works_On (essn, pno, hours)  Dependent (essn, depen_name, address, relationship, sex)</p> <p>Where:</p> <p>ssn: Primary key of Employee  dnumber: Primary key of Department and foreign key in Employee, Dept_Loc, and Project  pnumber: Primary key of Project  essn: Foreign key in Works_On and Dependent referencing Employee(ssn)  super_ssn: Foreign key referencing Employee(ssn)</p> <p>Write SQL Queries for the Following:</p> <ol style="list-style-type: none"> <li>1. Insert sample records into each table.</li> <li>2. Retrieve the names of employees who work on all the projects controlled by department number 3.</li> <li>3. Retrieve the names of employees who get the second highest salary.</li> <li>4. List the names of employees who have no dependents, in alphabetical order.</li> <li>5. Retrieve the number of employees and their average salary working in each department.</li> <li>6. Retrieve the SSN of all employees who work on at least one of the project numbers 1, 2, or 3.</li> <li>7. Retrieve the number of dependents for an employee named RAM.</li> <li>8. List the employees who do not work on any project.</li> <li>9. Retrieve the names of employees who live in the same address as their dependents.</li> <li>10. List the department names that have more than one location.</li> <li>11. Retrieve the names and addresses of all employees who work on more than two projects.</li> <li>12. Find the names of employees who earn more than their managers.</li> <li>13. Retrieve the highest salary paid in each department in descending order.</li> <li>14. Retrieve the names of employees who are paid the same salary as Anil.</li> <li>15. Retrieve the total number of hours worked by each employee on all projects.</li> </ol>

	16. Find the average salary of male and female employees separately.
2	<p>In recent years, most of the grocery items are available online, and people are increasingly using online platforms for purchasing. These platforms offer various discounts and benefits and help customers save time.</p> <p>To support such an application, a robust database system is needed to store and manage transaction data. The system must keep track of customers, agents, products, and all order transactions across various cities.</p> <p>Relational Schema</p> <p>Customer (cid, cname, city, phone)  Agent (aid, aname, city, phone)  Product (pid, pname, price, stock)  Orders (oid, cid, aid, odate, mode, status)  Order_Details (oid, pid, qty)</p> <p>Where:</p> <p>cid: Primary key of Customer and foreign key in Orders  aid: Primary key of Agent and foreign key in Orders  pid: Primary key of Product and foreign key in Order_Details  oid: Primary key of Orders and foreign key in Order_Details</p> <p>Perform the Following Operations and Write SQL Queries for the Following</p> <ol style="list-style-type: none"> <li>1. Add a column email with domain VARCHAR(50) to the Customer table.</li> <li>2. Insert some sample records into the each table.</li> <li>3. Update the phone number of the customer with cid = 3.</li> <li>4. Rename the column phone to mobile in the Agent table.</li> <li>5. Delete the customer whose cid is 5.</li> <li>6. Retrieve the names of customers who ordered more than 3 different products.</li> <li>7. List the names of agents who have delivered orders to customers in “Bangalore”.</li> <li>8. Display the total amount spent by each customer.</li> <li>9. Retrieve the names of products that are out of stock.</li> <li>10. List the agents who haven’t delivered any orders.</li> <li>11. Retrieve order IDs and product names for orders placed on '2025-08-01'.</li> <li>12. Display the number of orders delivered by each agent.</li> <li>13. Retrieve the highest priced product in each order.</li> <li>14. Find customers who haven’t placed any orders.</li> <li>15. List the most frequently ordered product.</li> </ol>
3	<p>The need for maintaining digital academic records is growing rapidly. Educational institutions such as schools, colleges, and universities have started storing students’ details such as name, age, marks, and enrolled courses in a structured database. This helps in tracking academic performance, course enrollment, and individual student progress efficiently. Design a Student Performance Management System database that stores the details of Students, their enrolled Courses, and the Marks they obtain using the following relational schema:</p> <p>Relational Schema</p> <p>Student (sid, sname, age, gender, course)  Course (course_id, course_name, instructor)  Marks (sid, course_id, marks)</p> <p>Where:</p> <p>sid: Primary key of Student and foreign key in Marks  course_id: Primary key of Course and foreign key in Marks</p> <p>Perform the following operations and write SQL Queries for the Following:</p> <ol style="list-style-type: none"> <li>1. Create the Student, Course, and Marks tables.</li> <li>2. Insert some sample records into the each table .</li> <li>3. Update the age of the student with sid = 3.</li> <li>4. Rename the column sname to student_name in the Student table.</li> <li>5. Delete the student whose sid = 5.</li> <li>6. Count the number of students from the Student table.</li> <li>7. Find the maximum marks obtained in any course.</li> </ol>

	<p>8. Find the minimum marks obtained in any course.</p> <p>9. Retrieve the average marks grouped by each course.</p> <p>10. Display the total number of students enrolled in each course.</p> <p>11. List the student names and marks in descending order of marks.</p> <p>12. Retrieve the sum of marks scored by students in each course.</p> <p>13. Display the minimum and maximum marks for each course.</p> <p>14. Find the average age of students.</p> <p>15. List all courses in alphabetical order along with the number of enrolled students.</p>
4	<p>Modern databases must ensure that concurrent access to data by multiple users does not lead to inconsistencies. Transaction processing plays a crucial role in maintaining data integrity and supporting operations like recovery and concurrency control. The system must handle multiple transactions while ensuring desirable properties like atomicity, consistency, isolation, and durability (ACID).</p> <p>Relational Schema</p> <ul style="list-style-type: none"> <li>● Accounts Fields: acc_no, cust_name, balance</li> <li>● Transactions Fields: trans_id, acc_no, trans_type, amount, trans_date</li> </ul> <p>Where Section</p> <p>acc_no: Foreign key in the Transactions table referencing Accounts trans_type: Can be 'deposit' or 'withdrawal'</p> <p>Each transaction must be processed atomically without leaving the database in an inconsistent state Multiple transactions can affect the same account and must be scheduled serializably</p> <p>Perform the Following Operations and Write Queries for the Following</p> <ol style="list-style-type: none"> <li>1. Create Accounts and Transactions tables using SQL.</li> <li>2. Insert sample records into both tables with a minimum of five transactions.</li> <li>3. Write a query to simulate a transaction: withdraw ₹500 from account acc_no = 101 and update balance accordingly.</li> <li>4. Show a schedule of two transactions accessing the same account and identify if the schedule is recoverable.</li> <li>5. Write a serial schedule and a non-serial schedule, and explain which one is conflict serializable.</li> <li>6. Identify a situation that causes cascading rollback and how recoverable schedules can avoid it.</li> <li>7. Implement COMMIT and ROLLBACK statements in SQL with proper examples.</li> <li>8. Explain with an example how a dirty read could happen if isolation is not enforced.</li> <li>9. Use SQL SET TRANSACTION command to assign different isolation levels and test behavior.</li> <li>10. Write an SQL query to check for concurrent updates on the same account and prevent them.</li> </ol>
5	<p>In recent years, most organizations maintain structured databases to manage customer information efficiently. To track changes in sensitive information such as salary, database systems often employ triggers for automatic auditing or monitoring.</p> <p>Relational Schema:</p> <p>CUSTOMERS (ID, NAME, AGE, ADDRESS, SALARY)</p> <p>Where:</p> <p>ID: Primary key of the CUSTOMERS table. SALARY: Numerical field representing the income of a customer. NAME, ADDRESS: Character fields storing customer information. AGE: Integer value representing the age of the customer.</p> <p>Perform the Following Operations and Write SQL Queries for the Following:</p> <ol style="list-style-type: none"> <li>1. Create the CUSTOMERS table with appropriate data types and constraints.</li> <li>2. Create a row-level trigger on the CUSTOMERS table that fires on INSERT, UPDATE, or DELETE operations: <ol style="list-style-type: none"> <li>a) On UPDATE, the trigger should display the salary difference between the old and new salary.</li> <li>b) On INSERT, the trigger should display the inserted customer's name and salary.</li> <li>c) On DELETE, the trigger should display the deleted customer's name and salary.</li> </ol> </li> <li>3. Insert five sample records into the CUSTOMERS table.</li> <li>4. Update the SALARY of the customer whose ID = 2.</li> </ol> <p>Delete the record of the customer whose ID = 4.</p>

6

In today's data-driven world, NoSQL databases like MongoDB offer flexible, schema-less storage suitable for handling large-scale and unstructured data. MongoDB is a widely used open-source NoSQL database that stores data in JSON-like documents with dynamic schemas, making it ideal for agile development and scalability.

To manage educational data efficiently, an institution has decided to store student records, including their ID, name, age, and enrolled course, in a NoSQL database system.

Relational Schema (Logical Document Structure)

Students

Fields: sid, sname, age, course, email, status

Where Section

sid: Primary key, uniquely identifies each student.

course: Stores the name of the enrolled course.

email: Optional field, can be added dynamically.

status: Indicates student status (e.g., "active").

All operations are to be performed on the students collection inside the mydb database.

Perform the Following Operations and Write Queries for the Following

Install MongoDB Community Edition on your system.

Create a database named mydb.

Create a collection named students.

1. Insert five documents into the students collection with the fields:
2. sid (student ID)
3. sname (student name)
4. age (student age)
5. course (enrolled course)
6. Display all the documents in the students collection.
7. Find a student document with sid = 3.
8. Update the course of the student whose sid = 2.
9. Delete a student document whose sid = 5.
10. Display all student names who are enrolled in the course "Data Science".
11. Sort the student documents by age in ascending order.
12. Count the number of students enrolled in "Full Stack Development".
13. Insert a new document with an additional field email into the students collection.
14. Update all documents to include a default status: "active" field.
15. Delete all students who are older than 25.
16. Drop the students collection from the database.

### Course Outcomes

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

CO1: Describe the basic elements of a database management system.

CO2: Apply the concepts of database management system for various applications

CO3: Analyse database concepts for a given problem.

CO4: Design SQL/NoSQL queries and conceptual models for database applications.

CO5: Demonstrate SQL and NoSQL queries to manipulate the data in the database.

**Assessment Details (both IAT and SEE)**

<b>Theory Component</b>	IAT-1 after completion 45 to 50% Syllabus	25 Marks
	IAT-2 after completion 95 to 100% Syllabus	25 Marks
	Average of two IATs	<b>25 Marks</b>
	<b>Total 25 Marks : Reduced to 15 Marks</b>	
	CCE-1	25 Marks
	CCE-2	25 Marks
	Average of two CCEs	<b>25 Marks</b>
	<b>Total 25 Marks : Reduced to 10 Marks</b>	
<b>Lab Component</b>	Lab Record and execution of programs	<b>15 Marks</b>
	<b>Lab Test at the end of 15<sup>th</sup> week</b>	<b>10 Marks</b>
	<b>Total</b>	<b>25 Marks</b>
<b>Grand Total of IAT Marks</b>		<b>50 Marks</b>
Obtaining 40% of marks in both theory and lab component is essential for appearing for SEE		

**Suggested Learning Resources:****Text Books:**

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill
3. P. J. Sadalage and M. Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Addison-Wesley Professional, 2013.

**Reference Books:**

1. An Introduction to Database Systems, C. J. Date, A. Kannan and S. Swamynathan, Pearson Education, Eighth Edition, 2009.
2. Database System Concepts, Abraham Silberschatz, Henry F. Korth and S. Sudarshan, McGraw-Hill Education (Asia), Fifth Edition, 2006.
3. Database Systems Concepts, Designs and Application, Shio Kumar Singh, Pearson Education, Second Edition, 2011.

**E - Resources:**

1. <https://www.javatpoint.com/dbms-tutorial>
2. <https://www.javatpoint.com/dbms-tutorial>
3. [Introduction to Database Management Systems \(youtube.com\)](#) by Neso Academy

## OBJECT-ORIENTED PROGRAMMING WITH C++

<b>Course Code</b>	<b>24CDT34</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours/Week (L:T:P: S)</b>	<b>3:0:0:0</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>40 hours Theory</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>03</b>	<b>Exam Hours</b>	<b>03</b>

### Course objectives:

#### This course will enable students to:

- CO1:** Understand and apply the fundamental concepts, syntax, and features of the C++ programming language.
- CO2:** Explain and use object-oriented programming principles and design techniques.
- CO3:** Design modular applications using classes, inheritance, and polymorphism.
- CO4:** Implement robust programs using exception handling and file I/O operations.
- CO5:** Utilize the Standard Template Library (STL) components such as containers, algorithms, and iterators to write generic and efficient C++ programs

### Teaching-Learning Process (General Instructions)

These are sample Strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic 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

**Introduction to C++ program-** Variables and Data types, Tokens: Keywords, Identifiers, Constants, Operators, Input and Output (cin, cout), Control statements, Functions in C++, Default arguments, Inline functions, Recursion. **08 Hours**

### Module – II

**Introduction to oops and Basic Principles of OOP : Encapsulation , Class and Object Basics –** Defining class, Access specifiers, Data members and member functions, **Constructors and Destructor ,Static Members and this Pointer**, Arrays of Objects, **Friend Function.** **08 Hours**

### Module – III

**Inheritance: Concept** of Inheritance: Base class, Derived class, Types of Inheritance. Access Control in Inheritance, Constructors and Destructors in Inheritance, Ambiguity resolution in multiple inheritance.  
**Polymorphism:** Compile-time Polymorphism (Function Overloading, Operator Overloading). Runtime Polymorphism: virtual functions, function overriding. Pure virtual functions and Abstract Classes. **08 Hours**

### Module – IV

**Exception Handling:** try, catch, throw, Multiple Catch Blocks, Standard Exception Classes, Nested Try Blocks, and Custom Exception Classes

**Introduction to File Handling in C++:** File stream classes, Opening and closing files, File modes, Reading and writing text files. File Error Handling **08 Hours**

**Module – V**

**Introduction to STL Components** – Containers, Algorithms, Iterators .**Sequence Containers** vector, list, deque. **Associative Containers** - set, map, multimap. **Derived Containers** –stack, queue, priority queue. **STL Algorithms** –sort, find, count, binary search. Iterators **08 Hours**

**Teaching-Learning Process for all modules** Chalk and board, Active Learning, PPT Based presentation, Video

**Course Outcomes:**

After studying this course, students will be able to

- CO1: Apply the syntax, data types, control structures, and functions of C++ to solve basic programming problems.
- CO2: Illustrate object-oriented design using classes, objects, constructors, and related features.
- CO3: Demonstrate the use of inheritance and polymorphism to develop modular and reusable software components.
- CO4: Develop robust programs using exception handling and file input/output operations.
- CO5: Explore the application of Standard Template Library (STL) components in generic programming.

**Assessment Details (both IAT and SEE)**

<b>Theory Component</b>	IAT-1 after completion 45 to 50% Syllabus	25 Marks
	IAT-2 after completion 95 to 100% Syllabus	25 Marks
	Average of two IATs	<b>25 Marks</b>
	CCE-1	25 Marks
	CCE-2	25 Marks
	Average of two CCEs	<b>25 Marks</b>
<b>Grand Total of IAT Marks (min marks 20 / 50)</b>		<b>50 Marks</b>
SEE conducted for 100 and scaled down to 50 (min marks 18/50)		<b>50 Marks</b>
IAT + SEE (min marks 40)		<b>100 Marks</b>

**Suggested Learning Resources:**

**Text Books:**

1. **E. Balagurusamy**, Object Oriented Programming with C++, Tata McGraw Hill Education, 7th Edition (or latest)

**Reference Books:**

1. **Bjarne Stroustrup**, The C++ Programming Language, Addison-Wesley, 4th Edition Authored by the creator of C++.
2. **Herbert Schildt**, C++: The Complete Reference, McGraw-Hill Education, 4th Edition
3. **Robert Lafore**, Object-Oriented Programming in C++, Sams Publishing

**Web links and Video Lectures (e-Resources):**

- <https://www.youtube.com/watch?v=ZzaPdXTrSb8&pp=0gcJCfwAo7VqN5tD>
- <https://www.youtube.com/watch?v=oOmbSpOzvYg&list=PLdo5W4Nhv31YU5Wx1dopka58teWP9aCee>
- NPTEL – C++ Programming
- GeeksforGeeks – C++ Tutorials  
<https://www.geeksforgeeks.org/c-plus-plus/>
- Programiz- C++  
<https://www.programiz.com/cpp-programming>

## C++ LAB

<b>Course Code</b>	<b>24CDL35</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours /Week(L:T:P)</b>	<b>(0:0:2)</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>12 Lab slots</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>01</b>	<b>Exam Hours</b>	<b>03</b>

### Course objectives:

#### This course will enable students to:

- CO1:** Understand and apply the fundamental concepts, syntax, and features of the C++ programming language.
- CO2:** Explain and use object-oriented programming principles and design techniques.
- CO3:** Design modular applications using classes, inheritance, and polymorphism.
- CO4:** Implement robust programs using exception handling and file I/O operations.
- CO5:** Utilize the Standard Template Library (STL) components such as containers, algorithms, and iterators to write generic and efficient C++ programs

### LIST OF LABORATORY PROGRAMS

1.	<p>Write a program to find all special numbers between 2 limits m and n(both inclusive). Assume that m and n are 2-digit numbers. A 2-digit number is said to be a special number if the sum of its digits and the products of its digits is equal to the number itself.</p> <p>For example, 19 is a special number. The digits in 19 are 1 and 9. The sum of the digits is 10 and the product of the digits is 9. <math>10+9 = 19</math>.</p> <p><b>Sample input:</b> 59.</p> <p><b>Sample output :</b> 59 is a Special Two-Digit Number</p>
2.	<p>Write a recursive program to reverse the digits of a number.</p> <ul style="list-style-type: none"> <li>• Get a number N from the user.</li> <li>• Use a <b>recursive method</b> to reverse the digits of the number.</li> <li>• Print the reversed number.</li> </ul> <p><b>Sample Input:</b></p> <p>1234</p> <p><b>Sample output:</b></p> <p>4321</p>
3.	<p>Write a program to create a class Money with two attributes <code>int rupee</code> and <code>int paisa</code>, including getter and setter methods, then create a main class to initialize two money amounts, add them properly (converting excess paisa into rupees), and print the total sum in rupee.paisa format.</p> <p><b>Sample Input</b></p> <p>50 85</p> <p>42 65</p> <p><b>Sample Output</b></p> <p>93.50</p>
4.	<p>Write a C++ program to create a class called <code>Distance</code> that stores a distance in kilometers and meters.</p> <p>The class should have two data members:</p> <ul style="list-style-type: none"> <li>• <code>int km</code> for kilometers</li> <li>• <code>int meters</code> for meters</li> </ul> <p>You must overload the constructors as follows:</p> <ol style="list-style-type: none"> <li>1. One constructor should accept two arguments — kilometers and meters.</li> <li>2. Another constructor should accept only one argument — meters. It should automatically</li> </ol>

	<p>convert this total meters into kilometers and meters.</p> <p>If the number of meters is <b>greater than or equal to 1000</b>, the class should <b>normalize</b> the values by converting extra meters into kilometers. For example, 1450 meters should become 1 km and 450 meters.</p> <p>The class should also include a member function <code>display()</code> that prints the distance in the format: In the <code>main()</code> function, prompt the user to:</p> <ul style="list-style-type: none"> <li>• Enter distance as kilometers and meters (two integers).</li> <li>• Enter another distance as only meters (one integer).</li> </ul> <p>Then create two objects using the two constructors and display the normalized distances. 1000 meters = 1 kilometer</p> <p><b>Sample Input:</b> Enter distance (km and meters): 2 1250 Enter distance in only meters: 1980</p> <p><b>Sample Output:</b> Distance = 3 km and 250 meters Distance = 1 km and 980 meters</p>
5.	<p>Write a C++ program to demonstrate <b>multiple inheritance</b>.</p> <ul style="list-style-type: none"> <li>• Create a class named Worker with protected data members: <ul style="list-style-type: none"> <li>◦ <code>int hoursWorked</code> and <code>int hourlyRate</code></li> <li>◦ A member function to calculate salary as <math>\text{salary} = \text{hoursWorked} \times \text{hourlyRate}</math></li> </ul> </li> <li>• Create another class named Bonus with: <ul style="list-style-type: none"> <li>◦ A protected data member <code>int bonusAmount</code></li> </ul> </li> <li>• Create a third class named FinalSalary which is derived from both Worker and Bonus.</li> <li>• In the FinalSalary class: <ul style="list-style-type: none"> <li>◦ Calculate the total salary as <math>\text{finalSalary} = \text{salary} + \text{bonusAmount}</math></li> <li>◦ Display the worker's salary, bonus, and final salary.</li> </ul> </li> </ul> <p>In the <code>main()</code> function:</p> <ul style="list-style-type: none"> <li>• Accept input from the user for hours worked, hourly rate, and bonus amount.</li> <li>• Create an object of FinalSalary and display the result.</li> </ul> <p><b>Sample Input:</b> Enter hours worked: 40 Enter hourly rate: 200 Enter bonus: 1000</p> <p><b>Sample Output:</b> Salary: 8000 Bonus: 1000 Final Salary: 9000</p>
6.	<p>Write a C++ program to demonstrate <b>runtime polymorphism</b> using virtual functions.</p> <ul style="list-style-type: none"> <li>• Create a base class Employee with a <b>virtual function</b> <code>displaySalary()</code>.</li> <li>• Create two derived classes: <ul style="list-style-type: none"> <li>◦ Manager</li> <li>◦ Engineer</li> </ul> </li> <li>• Each derived class should have a specific salary and must <b>override</b> the <code>displaySalary()</code> function to print the respective salary.</li> </ul> <p>In the <code>main()</code> function:</p> <ul style="list-style-type: none"> <li>• Create objects of Manager and Engineer classes.</li> <li>• Call the <code>displaySalary()</code> function on each object to display their salary.</li> </ul> <p><b>Sample Input:</b> Enter Manager Salary: 80000</p>

	<p>Enter Engineer Salary: 50000</p> <p><b>Sample Output</b>  Manager Salary: Rs. 80000  Engineer Salary: Rs. 50000</p>
7.	<p>Write a program to check whether a person is eligible for a driving license.</p> <ul style="list-style-type: none"> <li>• If the person's age is less than 18, throw an exception and print "<b>Invalid Age</b>" and exit the program.</li> <li>• If the score is less than 40, throw an exception and print "<b>You should get at least 40 marks</b>" and exit the program.</li> <li>• Otherwise, print the <b>age</b>, <b>score</b>, and the message "<b>Passed</b>".</li> </ul> <p><b>Sample Input:</b>  12 50</p> <p><b>Sample Output:</b>  12 50  Invalid Age</p>
8.	<p>Write a C++ program to create a text file, check file created or not, if created it will write some text into the file and then read the text from the file.</p>
9.	<p>Write a C++ program that uses <b>STL vector</b> to perform the following task:  You are given an array of integers stored in a <code>vector&lt;int&gt; arr</code> and an integer <code>k</code>. Your task is to write a function that returns a new <code>vector&lt;int&gt;</code> containing only those elements from the original vector which are <b>less than k</b>.</p> <ul style="list-style-type: none"> <li>• The <b>relative order</b> of elements in the result must be the <b>same</b> as in the input vector.</li> <li>• If no such element exists, the function should return an <b>empty vector</b>.</li> </ul> <p>You must use <code>vector</code> from the Standard Template Library (STL).</p> <p><b>Sample Input:</b>  5  5 3 6 1 3  4</p> <p><b>Sample Output:</b>  3 1 3</p>
10.	<p>Write a C++ program to manage a student record system using a <code>map&lt;int, string&gt;</code>, where:</p> <ul style="list-style-type: none"> <li>• The <b>key</b> represents the student's <b>roll number</b></li> <li>• The <b>value</b> represents the student's <b>name</b></li> </ul> <p>The program should do the following:</p> <ol style="list-style-type: none"> <li>1. Ask the user to enter <b>how many student records</b> they want to store. <ul style="list-style-type: none"> <li>○ For each student, accept the roll number and name from the user.</li> <li>○ Insert the data into a <code>map&lt;int, string&gt;</code>.</li> </ul> </li> <li>2. After storing all the records, <b>display all student entries</b> using an <b>iterator</b>. <ul style="list-style-type: none"> <li>○ Each entry should print the roll number and corresponding name.</li> </ul> </li> <li>3. Then, ask the user to <b>enter a roll number to search</b>. <ul style="list-style-type: none"> <li>○ If the roll number is found in the map, print the student's name.</li> <li>○ If not found, display an appropriate message.</li> </ul> </li> </ol> <p><b>Sample Input:</b>  Enter number of students: 6  Ravi 101  Meena 102  Arjun 103  Kavya 104  Enter roll number to search: 102</p>

**Sample Output:**

Student found: Meena

**Course Outcomes:**

After studying this course, students will be able to

- CO1: Apply the syntax, data types, control structures, and functions of C++ to solve basic programming problems.
- CO2: Illustrate object-oriented design using classes, objects, constructors, and related features.
- CO3: Demonstrate the use of inheritance and polymorphism to develop modular and reusable software components.
- CO4: Develop robust programs using exception handling and file input/output operations.
- CO5: Explore the application of Standard Template Library (STL) components in generic programming.

**Text Books:**

2. **E. Balagurusamy**, Object Oriented Programming with C++, Tata McGraw Hill Education, 7th Edition (or latest)

**Reference Books:**

1. **Bjarne Stroustrup**, The C++ Programming Language, Addison-Wesley, 4th Edition Authored by the creator of C++.
2. **Herbert Schildt**, C++: The Complete Reference, McGraw-Hill Education, 4th Edition
3. **Robert Lafore**, Object-Oriented Programming in C++, Sams Publishing

**Web links and Video Lectures (e-Resources):**

- <https://www.youtube.com/watch?v=ZzaPdXTrSb8&pp=0gcJCfwAo7VqN5tD>
- <https://www.youtube.com/watch?v=oOmbSpOzvYg&list=PLdo5W4Nhhv31YU5Wx1dopka58teWP9aCee>
- NPTEL – C++ Programming
- GeeksforGeeks – C++ Tutorials  
<https://www.geeksforgeeks.org/c-plus-plus/>
- Programiz- C++  
<https://www.programiz.com/cpp-programming>

**Assessment Details(both IAT and SEE)**

Continuous Internal Assessment of Laboratory/Practical Courses		
Lab Test 1	Lab Test 2	Lab Records
15 marks	15 marks	20 marks
Semester End Examination(SEE)		50 marks

**Conduct of Practical Examination:****Experiment distribution :**

**For laboratories having only one part:** Students are allowed to pick one experiment from the lot with equal opportunity.

**For laboratories having PART A and PART B:** Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (Need to change in accordance with university regulations)
  - a) For laboratories having only one part → Procedure + Execution + Viva-Voce:  
15+70+15 = 100 Marks
  - b) For laboratories having PART A and PART B
    - i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
    - ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks

## IT INFRASTRUCTURE AND MANAGEMENT

<b>Course Code</b>	<b>24CDT361</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours/Week (L:T:P: S)</b>	<b>3:0:0:0</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>40</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>03</b>	<b>Exam Hours</b>	<b>03</b>

### Course Objectives:

#### The Student will:

1. Understanding the role of IT infrastructure with its functions and services.(Understanding)
2. Recognize the research, reporting and presentation approach using the latest ICT tools.
3. Study the combination of the technical and management issues in contemporary infrastructure management.
4. Learn the concepts related with Deadlock to solve Problems.
5. Familiarize the Protection and Security Mechanism in Operating System.

### Teaching-Learning Process (General Instructions)

These are sample Strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic 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

**Infrastructure management overview:** Introduction, IS components, Services of IT infrastructure, welfare of IT, Roles and responsibilities, challenges.

**Organizing for Infrastructure management:** IT infrastructure design factors, model of IT management, Elucidation methods, Documentation. **08 Hours**

### Module – II

**Staffing for system management:** Introduction, Determining Required Skill Sets and Skill Levels Assessing the Skill Levels of Current Onboard Staff.

**Customer Service:** Introduction, Key Elements of Good Customer Service: Key Customers, Identifying Key Services, Identifying Key Processes that Support Key Services, Key Suppliers, Integrating the Key Elements of Good Customer Service, Cardinal Sins that Undermine Good Customer Service. **08 Hours**

### Module – III

**Performance and Tuning:** Introduction, Performance and Tuning Applied to the Five Major Resource Environments: Server Environment, Disk Storage Environment, Database Environment, Network Environment and Desktop Computer Environment.

**Problem management:** The role of service desk, segregating and integrating service desk, Developing a Problem Management Process, client issues with problem management.

**08 Hours**

**Module – IV**

**Storage Management:** Storage Management Capacity, Storage Management Performance, Storage Management Reliability, Storage Management Recoverability.

**Network Management:** Key Decisions about Network Management, business IT networks and components, digital transmission, IS vulnerabilities and threats.

**08 Hours**

**Module – V**

**Strategic Security:** Introduction, Developing a Strategic Security Process, IT Strategic planning process, Tools & methodologies of IT strategic planning, Business system planning approach.

**Facilities management:** Introduction, Major Elements of Facilities Management, Tips to improve facilities management process.

**08 hours**

**Course outcomes:**

**The Student will be able to:**

CO1: Investigate, critically analyze and evaluate the impact of new and current ICT services to an organization.

CO2: Demonstrate the technical and communications skills that contribute to the operation of ICT services in an organization.

CO3: Reflect critically on the role of an enterprise architect in an organization.

CO4: Gain Knowledge on theoretical, technical and management issues that deliver ICT services to an organization.

CO5: Analyze how effective IT Infrastructure Management requires strategic planning with alignment from both the IT and business perspectives in an organization.

**Assessment Details (both IAT and SEE)**

<b>Theory Component</b>	IAT-1 after completion 45 to 50% Syllabus	25 Marks
	IAT-2 after completion 95 to 100% Syllabus	25 Marks
	Average of two IATs	<b>25 Marks</b>
	CCE-1	25 Marks
	CCE-2	25 Marks
	Average of two CCEs	<b>25 Marks</b>
<b>Grand Total of IAT Marks (min marks 20 / 50)</b>		<b>50 Marks</b>
SEE conducted for 100 and scaled down to 50 ( <b>min marks 18/50</b> )		<b>50 Marks</b>
IAT + SEE (min marks 40)		<b>100 Marks</b>

**Suggested Learning Resources:**

**Text Books:**

1. Rich Schiesser, IT Systems Management, Pearson Second Edition.
2. IT for Management Turban Volonino.

**Reference Books:**

1. E Turban, E Mclean and James Wetherbe, —Information Technology for Management
2. Kenneth C Laudon, Jane P Laudon, —Management Information Systems
3. Roger S Pressman, —Software Engineering: A Practitioner’s Approach
4. James A O’Brien, —Management Information Systems
5. Walker Royce, — Software Project Management: A Unified Framework

**E - Resources:**

1. <https://www.scribd.com/doc/45079962/IT-Infrastructure-Management>
2. <https://www.scribd.com/document/509694935/IT-Infrastructure-Management-eI9RGuDM0m>

## BUSINESS PROCESS FUNDAMENTALS

<b>Course Code</b>	<b>24CDT362</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours/Week (L:T:P: S)</b>	<b>3:0:0:0</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>40</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>03</b>	<b>Exam Hours</b>	<b>03</b>

### Course Learning Objectives:

This course will enable students to:

1. Understand the basics of business and economy.
2. Learn the basics ethics of entrepreneurship and how to start a business.
3. Understand the roles of ownership, management and leadership.
4. Learn how to design an organization and its operations.
5. Understand the roles and management of human resource in an organization.

### Teaching-Learning Process (General Instructions)

These are sample Strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic 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

#### Teamwork & Economics

**Teamwork in Business**, The Foundation of Business – Introduction, Getting Down to Business, Functional Areas of Business, External Forces that Influence Business Activities.

**Economics and Business** - What is Economics? Perfect Competition and Supply and Demand, Monopolistic Competition, Oligopoly, and Monopoly, Measuring the Health of the Economy, Government's Role in Managing the Economy.

**Textbook1: Chapter 1 & 2**

**08 Hours**

**Teaching Learning Methodology:** Chalk & Talk, Demo using Python IDE

### Module – II

#### Ethics and Entrepreneurship

**Ethics** – Introduction, What is Business Ethics? Identifying Ethical Issues and Dilemmas, Corporate Social Responsibility, Ethical Organizations, The Individual Approach to Ethics.

**Entrepreneurship** - The Nature of Entrepreneurship, The Importance of Small Business to the U.S. Economy, What Industries Are Small Businesses In? Advantages and Disadvantages of Business Ownership, Starting a Business, Why Some Businesses Fail and Where to Get Help.

**Textbook1: Chapter 3, 4, 5 & 6**

**08 Hours**

**Teaching Learning Methodology:** Chalk & Talk, Problem based learning:  
[https://onlinecourses.nptel.ac.in/noc19\\_ee53/](https://onlinecourses.nptel.ac.in/noc19_ee53/)

**Module – III**

**Ownership, Management and Leadership**

**Ownership** - The Ice Cream Men , Factors to Consider, Mergers and Acquisitions.

**Management and Leadership** - Noteworthy Management, What Do Managers Do?, Planning, Leading, Controlling, Managerial Skills, Applying Your Skills at Notes-4-You.

**Textbook1: Chapter 7**

**08 Hours**

**Teaching Learning Methodology:** Chalk & Talk, Problem based learning:  
[https://onlinecourses.nptel.ac.in/noc19\\_ee53/](https://onlinecourses.nptel.ac.in/noc19_ee53/)

**Module – IV**

**Organizational structures and Operation Management**

**Organizational structure** – Organizing, Organizational Structure: How companies do the job done.

**Operation Management** - The Challenge: Producing Quality Jetboards, Operations Management in Manufacturing, Managing the Production Process in a Manufacturing Company, Graphical Tools: Gantt and PERT Charts, The Technology of Goods Production, Operations Management for Service Providers, Producing for Quality.

**Textbook1: Chapter 8 & 9**

**08 Hours**

**Teaching Learning Methodology:** Chalk & Talk, Problem based learning:  
[https://onlinecourses.nptel.ac.in/noc22\\_ge04/](https://onlinecourses.nptel.ac.in/noc22_ge04/)

**Module – V**

**Human resources and Motivating employees**

**Motivating Employees** – Motivation, Hierarchy of Needs Theory, Two-Factor Theory, Expectancy Theory, Equity Theory.

**Managing Human Resources** - Human Resource Management, Developing Employees, What Makes a Great Place to Work? Compensation and Benefits, Performance Appraisal.

**Textbook1: Chapter 10 & 11**

**08 hours**

**Teaching Learning Methodology:** Chalk & Talk, Problem based learning:  
[https://onlinecourses.nptel.ac.in/noc22\\_ge04/](https://onlinecourses.nptel.ac.in/noc22_ge04/)

**Course Outcomes:**

**On completion of this course, the students will be able to:**

- CO1: Design the basics of any business
- CO2: Design the rules and social responsibility of an organization.
- CO3: Develop the roles of the interrelated functions of management.
- CO4: Construct and manage an organization.
- CO5: Utilize the human resources effectively by motivating the employees.

**Assessment Details (both IAT and SEE)**

<b>Theory Component</b>	IAT-1 after completion 45 to 50% Syllabus	25 Marks
	IAT-2 after completion 95 to 100% Syllabus	25 Marks
	Average of two IATs	<b>25 Marks</b>
	CCE-1	25 Marks
	CCE-2	25 Marks
	Average of two CCEs	<b>25 Marks</b>
<b>Grand Total of IAT Marks (min marks 20 / 50)</b>		<b>50 Marks</b>
SEE conducted for 100 and scaled down to 50 (min marks 18/50)		<b>50 Marks</b>
IAT + SEE (min marks 40)		<b>100 Marks</b>

**Suggested Learning Resources:****Textbooks:**

1. Stephen J. Skripak, Fundamentals of Business, Pamplin college of Business and Virginia Tech Libraries, 2016.

**Reference:**

1. S.S. Kanaka, Entrepreneurial Development, S-Chand Fourth Edition.
2. Robert D. Hisrich and Michael P. Peters, Entrepreneurship, McGraw – Hill Publication.
3. Poornima M. Charantimath, Entrepreneurship Development Small Business Enterprises, Pearson Education.
4. Thomas. W. Zimmerer & Norman. M. Scarborough, Essentials of Entrepreneurship and Small Business Management, PHI.

**E-Resources:**

1. Lee Angelelli (1994). "Steve Paul Jobs." Retrieved from: <http://ei.cs.vt.edu/~history/Jobs.html>
2. Warren E. Buffet and Carol Loomis (2003). "America's Growing Trade Deficit Is Selling The Nation Out From Under Us. Here's A Way To Fix The Problem--And We Need To Do It Now." Fortune. November 10, 2003. Retrieved June 9, 2016 from: [http://archive.fortune.com/magazines/fortune/fortune\\_archive/2003/11/10/352872/index.htm](http://archive.fortune.com/magazines/fortune/fortune_archive/2003/11/10/352872/index.htm)
3. Coca Cola Company (2016). "Our Company: Vision, Mission, and Values." Cocacola.com. Retrieved from: <http://www.coca-colacompany.com/our-company/mission-vision-values>
4. Johnson and Johnson (2016). "Company Structure." Retrieved from: <http://www.jnj.com/about-jnj/company-structure>
5. Burger King (2016). "About Us." Burger King Website: bk.com. Retrieved from: <http://www.bk.com/about-bk>
6. Starbucks (2016). "Working at Starbucks." Starbucks.com. Retrieved from: <http://www.starbucks.com/careers/working-at-starbucks>
7. Fortune (2007). "100 Top MBA Employers." Fortune. Retrieved from: [http://archive.fortune.com/magazines/fortune/mba100/2007/full\\_list/index.html](http://archive.fortune.com/magazines/fortune/mba100/2007/full_list/index.html)

**Activity Based Learning (Suggested Activities in Class) / Practical Based learning**

- Real world problem solving by giving group projects.
- Group discussion on finding suitable learning algorithm for the problem.
- Seminar on advanced techniques by the students.

## SUPPLY CHAIN MANAGEMENT

<b>Course Code</b>	<b>24CDT363</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours/Week (L:T:P: S)</b>	<b>3:0:0:0</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>40</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>03</b>	<b>Exam Hours</b>	<b>03</b>

### Course Objectives:

The objectives of this course are

1. To provide Knowledge on logistics and supply chain management
2. To enable them in designing the distribution network
3. To train the students in knowing the supply chain Analysis
4. Impart knowledge on Dimensions of logistic
5. To know the recent trends in supply chain management

### Teaching-Learning Process (General Instructions)

These are sample Strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic 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

**Introduction to Supply Chain Management:** Supply chain - objectives - importance - decision phases - process view competitive and supply chain strategies - achieving strategic fit – supply chain drivers - obstacles – framework – facilities -inventory-transportation-information-sourcing-pricing.

**08 Hours**

### Module – II

**Designing the distribution network:** Role of distribution - factors influencing distribution - design options - e-business and its impact distribution networks in practice –network design in the supply chain - role of network -factors affecting the network design decisions modeling for supply chain. Role of transportation - modes and their performance – transportation infrastructure and policies - design options and their trade-offs tailored transportation.

**08 Hours**

### Module – III

**Supply Chain Analysis:** Sourcing - In-house or Outsource - 3rd and 4th PLs - supplier scoring and assessment, selection - design collaboration - Procurement process - Sourcing planning and analysis. Pricing and revenue management for multiple customers, perishable products, seasonal demand, bulk and spot contracts.

**08 Hours**

### Module – IV

**Dimensions of Logistics:** A macro and micro dimension - logistics interfaces with other areas - approach to analyzing logistics systems - logistics and systems analysis - techniques of logistics system analysis - factors affecting the cost and importance of logistics. Demand Management and Customer Service Outbound to customer logistics systems - Demand Management –Traditional Forecasting CPFRP - customer service - expected cost of stock outs - channels of distribution.

**08 Hours**

### Module – V

**Recent Trends in Supply Chain Management-Introduction,** New Developments in Supply Chain Management, Outsourcing Supply Chain Operations, Co-Maker ship, The Role of E- Commerce in Supply Chain Management, Green Supply Chain Management, Distribution Resource Planning, World Class Supply Chain Management.

**08 Hours**

#### Course Outcomes:

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

CO1: Analyze and design effective strategies for managing and optimizing supply chains to improve operational efficiency, competitiveness, and overall business performance.

CO2: Design and optimize distribution networks and transportation strategies, considering key factors, design options, and the impact of e-business on supply chain performance.

CO3: Analyze sourcing, procurement, and pricing strategies, including supplier selection, collaboration, and revenue management for various customer demands and product types.

CO4: Analyze logistics systems, apply demand management techniques, and optimize customer service and distribution to improve efficiency and reduce costs.

CO5: Evaluate and apply recent trends in supply chain management, including digitalization, sustainability, and innovative technologies, to enhance supply chain performance.

#### Assessment Details (both IAT and SEE)

<b>Theory Component</b>	IAT-1 after completion 45 to 50% Syllabus	25 Marks
	IAT-2 after completion 95 to 100% Syllabus	25 Marks
	Average of two IATs	<b>25 Marks</b>
	CCE-1	25 Marks
	CCE-2	25 Marks
	Average of two CCEs	<b>25 Marks</b>
<b>Grand Total of IAT Marks (min marks 20 / 50)</b>		<b>50 Marks</b>
SEE conducted for 100 and scaled down to 50 (min marks 18/50)		<b>50 Marks</b>
IAT + SEE (min marks 40)		<b>100 Marks</b>

#### Suggested Learning Resources:

##### Text Books:

1. Sunil Chopra and Peter Meindl, Supply Chain Management – “Strategy, Planning and Operation”, 3<sup>rd</sup> Edition, Pearson/PHI,2007.
2. Supply Chain Management by Janat Shah Pearson Publication 2008.

##### Reference Books:

1. A Logistic approach to Supply Chain Management – Coyle, Bardi, Longley, Cengage Learning, 1/e
2. Donald J Bowersox, Dand J Closs, M Bixby Coluper, “Supply Chain Logistics Management”, 2<sup>nd</sup> edition, TMH,2008.
3. Wisner, Keong Leong and Keah-Choon Tan, “Principles of Supply Chain Management A Balanced Approach”, Cengage Learning,1/e
4. David Simchi-Levi et al, “Designing and Managing the Supply Chain” –Concepts

## HUMAN COMPUTER INTERACTION

<b>Course Code</b>	<b>24CDT364</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours/Week (L:T:P: S)</b>	<b>3:0:0:0</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>40</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>03</b>	<b>Exam Hours</b>	<b>03</b>

### Course Learning Objectives

1. To learn the foundations of Human Computer Interaction.
2. To become familiar with the design technologies for individuals and persons with disabilities.
3. To be aware of mobile HCI.
4. To learn the guidelines for user interface.
5. Develop meaningful user interface.

### Teaching-Learning Process (General Instructions)

These are sample Strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic 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

#### Introduction:

Importance of user Interface – definition, importance of good design. Benefits of good design. A brief history of Screen design. The graphical user interface – popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user – Interface popularity, characteristics-Principles of user interface.

**08 Hours**

### Module – II

#### Design Process & Screen Designing:

Design process – Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, understanding business junctions.

Screen Designing: Design goals – Screen planning and purpose, organizing screen elements, ordering of screen data and content – screen navigation and flow – Visually pleasing composition – amount of information – focus and emphasis – presentation information simply and meaningfully – information retrieval on web – statistical graphics – Technological consideration in interface design.

**08 Hours**

### Module – III

**Windows and Components**  
 Windows – New and Navigation schemes selection of window, selection of devices based and screen-based controls. Components – text and messages, Icons and increases – Multimedia, colors, uses problems, choosing colors.

**08 Hours**

**Module – IV**

**HCI in the Software Process**

HCI in the software process, The software life cycle Usability engineering Iterative design and prototyping Design Focus: Prototyping in practice Design rationale Design rules Principles to support usability Standards Golden rules and heuristics HCI patterns Evaluation techniques, Goals of evaluation, Evaluation through expert analysis, Evaluation through user participation, Choosing an evaluation method. Universal design, Universal design principles Multi-modal interaction.

**08 Hours**

**Module – V**

**Cognitive Models**

Cognitive models Goal and task hierarchies Design Focus: GOMS saves money Linguistic models The challenge of display-based systems Physical and device models Cognitive architectures Ubiquitous computing and augmented realities Ubiquitous computing applications research Design Focus: Ambient Wood – augmenting the physical Virtual and augmented reality Design Focus: Shared experience Design Focus: Applications of augmented reality Information and data visualization Design Focus: Getting the size right.

**08 Hours**

**Teaching-Learning Process**

Chalk and board, Active Learning, Demonstration, Web content, Case Study

**Course Outcomes**

On completion of this course, the students will be able to,

CO1: Design effective dialog for HCI.

CO2: Design effective HCI for individuals and persons with disabilities.

CO3: Explain the HCI implications for designing multimedia/ ecommerce/ e-learning Web sites.

CO4: Assess the importance of user feedback.

CO5: Design and develop meaningful user interface.

**Assessment Details (both IAT and SEE)**

<b>Theory Component</b>	IAT-1 after completion 45 to 50% Syllabus	25 Marks
	IAT-2 after completion 95 to 100% Syllabus	25 Marks
	Average of two IATs	<b>25 Marks</b>
	CCE-1	25 Marks
	CCE-2	25 Marks
	Average of two CCEs	<b>25 Marks</b>
<b>Grand Total of IAT Marks (min marks 20 / 50)</b>		<b>50 Marks</b>
SEE conducted for 100 and scaled down to 50 ( <b>min marks 18/50</b> )		<b>50 Marks</b>
IAT + SEE (min marks 40)		<b>100 Marks</b>

**Suggested Learning Resources:**

**Textbooks:**

1. Wilbert O. Galitz, The Essential Guide to user Interface Design: An Introduction to GUI Design Principles and Techniques, Wiley, Second Edition 2002. (Module I, II, III)
2. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, Human Computer Interaction, 3rd Edition, Pearson Education, 2005 (Module IV, V)

**Reference:**

1. Andrew Monk, Fundamentals of Human Computer Interaction, 1st Edition, Academic Press, 2014.
2. Ben Shneiderman, Catherine Plaisant, Maxine S. Cohen, Steven M. Jacobs, Designing the User Interface: Strategies for Effective Human-Computer Interaction, 5th Edition, Pearson Education Asia Pearson, 2009
3. Brian Fling, Mobile Design and Development, First Edition, O'Reilly Media Inc., 2009

4. Bill Scott and Theresa Neil, Designing Web Interfaces, First Edition, O'Reilly, 2009.

**Weblinks and Video Lectures (e-Resources):**

1. <https://nptel.ac.in/courses/106103115>
2. <https://archive.nptel.ac.in/courses/106/106/106106177/>
3. [https://www.tutorialspoint.com/human\\_computer\\_interface/index.htm](https://www.tutorialspoint.com/human_computer_interface/index.htm)

## SOCIAL CONNECT AND RESPONSIBILITY

<b>Course Code</b>	<b>24UHV37</b>	<b>CIE Marks</b>	<b>100</b>
<b>Teaching Hours/Week (L:T:P: S)</b>	<b>0:0:2:0</b>	<b>SEE Marks</b>	<b>--</b>
<b>Total Hours of Pedagogy</b>	<b>30 hours Practical Session</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>01</b>	<b>Exam Hours</b>	<b>--</b>

**Course objectives: The course will enable the students to:**

1. Provide a formal platform for students to communicate and connect to the surrounding.
2. create a responsible connection with the society.
3. Understand the community in general in which they work.
4. Identify the needs and problems of the community and involve them in problem –solving.
5. Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.
6. Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.

**General Instructions - Pedagogy :**

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the activities will develop students' theoretical and applied social and cultural skills.
2. State the need for activities and its present relevance in the society and Provide real-life examples.
3. Support and guide the students for self-planned activities.
4. You will also be responsible for assigning homework, grading assignments and quizzes, and documenting students' progress in real activities in the field.
5. Encourage the students for group work to improve their creative and analytical skills.

**Contents :**

The course is mainly activity-based that will offer a set of activities for the student that enables them to connect with fellow human beings, nature, society, and the world at large.

The course will engage students for interactive sessions, open mic, reading group, storytelling sessions, and semester-long activities conducted by faculty mentors.

In the following a set of activities planned for the course have been listed:

### Module – I

**Plantation and adoption of a tree:**

Plantation of a tree that will be adopted for four years by a group of BE / B.Tech students. (ONE STUDENT ONE TREE) They will also make an excerpt either as a documentary or a photo blog describing the plant's origin, its usage in daily life, its appearance in folklore and literature. **06 Hours**

### Module – II

**Heritage walk and crafts corner:**

Heritage tour, knowing the history and culture of the city, connecting to people around through their history, knowing the city and its craftsman, photo blog and documentary on evolution and practice of various craft forms. **06 Hours**

### Module – III

**Organic farming and waste management:**

Usefulness of organic farming, wet waste management in neighboring villages, and implementation in the campus. **06 Hours**

### Module – IV

**Water conservation:**

Knowing the present practices in the surrounding villages and implementation in the campus,

documentary or photo blog presenting the current practices.	<b>06 Hours</b>
<b>Module – V</b>	
<b>Food fest:</b> City's culinary practices, food lore, and indigenous materials of the region used in cooking/ food festivals.	<b>06 Hours</b>
<b>Course Outcomes:</b> The students will be able to : CO1: Create a responsible connection with the society to address real-world societal challenges, including issues of corporate social responsibility (CSR) and sustainability. CO2: Identify the needs and problems of the community and involve them in problem –solving. CO3: Implement practices that promote sustainability in personal and professional life, contributing to long-term societal welfare. CO4: Work collaboratively in teams to solve complex social problems, demonstrating teamwork, empathy, and collective responsibility. CO5: Demonstrate the implemented idea through presentation and report.	
<b>E-Resources:</b> <a href="https://ffreedom.com/english/about-app?gad_source=1&amp;gclid=CjwKCAjw74e1BhBnEiwAbqOAjBOnuaO-a6CMVTjwubTiz2e13DWkkZ7ZpwX8PELXz5NRhe2a5cG2dBoCjwQQA vD_BwE">https://ffreedom.com/english/about-app?gad_source=1&amp;gclid=CjwKCAjw74e1BhBnEiwAbqOAjBOnuaO-a6CMVTjwubTiz2e13DWkkZ7ZpwX8PELXz5NRhe2a5cG2dBoCjwQQA vD_BwE</a> <a href="https://prasarbharati.gov.in/dd-kisan-homepage/">https://prasarbharati.gov.in/dd-kisan-homepage/</a>	
<b>ACTIVITIES:</b> Jamming session, open mic, and poetry: Platform to connect to others. Share the stories with others. Share the experience of Social Connect. Exhibit the talent like playing instruments, singing, one-act play, art-painting, and fine art.	
<b>PEDAGOGY:</b> The pedagogy will include interactive lectures, inspiring guest talks, field visits, social immersion, and a course project. Applying and synthesizing information from these sources to define the social problem to address and take up the solution as the course project, with your group. Social immersion with NGOs/social sections will be a key part of the course. Will all lead to the course project that will address the needs of the social sector?	
<b>COURSE TOPICS:</b> The course will introduce social context and various players in the social space, and present approaches to discovering and understanding social needs. Social immersion and inspiring conversational will culminate in developing an actual, idea for problem-based intervention, based on an in-depth understanding of a key social problem.	
<b>Duration:</b> A total of 40 - 50 hrs engagement per semester is required for the 3rd semester of the B.E. /B.Tech. program. The students will be divided into groups. Each group will be handled by faculty mentor. Faculty mentor will design the activities (particularly Jamming sessions open mic ,and poetry) Faculty mentors has to design the evaluation system as per VTU guidelines of scheme & syllabus.	
<b>Guideline for Assessment Process:</b> <b>Continuous Internal Evaluation (CIE):</b> After completion of the course, the student shall prepare, with daily diary as reference, a comprehensive report in consultation with the mentor/s to indicate what he has observed and learned in the social connect period. The report should be signed by the mentor. The report shall be evaluated on the basis of the following criteria and/or other relevant criteria pertaining to the activity completed. Marks allotted for the diary are out of 50. Planning and scheduling the socialconnect Information/Data collected during the social connect Analysis of the information/data and report writing Considering all above points allotting the marks as mentioned below <b>Excellent : 80 to 100</b> <b>Good : 60 to 79</b> <b>Satisfactory : 40 to 59</b>	

**Unsatisfactory and fail : <39**

**Special Note:**

**NO SEE – Semester End Exam – Completely Practical and activities based evaluation**

**Pedagogy – Guidelines:**

**It may differ depending on local resources available for the study as well as environment and climatic differences, location and time of execution.**

Sl No	Topic	Group size	Location	Activity execution	Reporting	Evaluation Of the Topic
1.	<b>Plantation and adoption of a tree:</b>	May be individual or team	Farmers land/ parks / Villages / roadside/ community area / College campus etc.....	Site selection /proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
2.	<b>Heritage walk and crafts corner:</b>	May be individual or team	Temples / monumental places / Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers/ campus etc.....	Site selection /proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
3.	<b>Organic farming and waste management:</b>	May be individual or team	Farmers land / parks / Villages visits / roadside/ community area / College campus etc.....	Group selection / proper consultation / Continuous monitoring / Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
4.	<b>Water conservation: &amp; conservation techniques</b>	May be individual or team	Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers / campus etc.....	site selection / proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
5.	<b>Food fest: Practices in society</b>	May be individual or team	Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers/ campus etc.....	Group selection / proper consultation / Continuous monitoring / Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty

### Plan of Action (Execution of Activities )

Sl.NO	Practice Session Description
1	Lecture session in field to start activities
2	Students Presentation on Ideas
3	Commencement of activity and its progress
4	Execution of Activity
5	Execution of Activity
6	Execution of Activity

7	Execution of Activity
8	Case study based Assessment, Individual performance
9	Sector/ Team wise study and its consolidation
10	Video based seminar for 10 minutes by each student At the end of semester with Report.

- Each student should do activities according to the scheme and syllabus.
- At the end of semester student performance has to be evaluated by the faculty for the assigned activity progress and its completion.
- At last consolidated report of all activities from 1<sup>st</sup> to 5<sup>th</sup>, compiled report should be submitted as per the instructions and scheme.

Assessment Details for CIE (both CIE and SEE)

Weightage	CIE – 100%	<ul style="list-style-type: none"> <li>• Implementation strategies of the project (NSS work).</li> <li>• The last report should be signed by NSS Officer, the HOD and principal.</li> <li>• At last report should be evaluated by the NSS officer of the institute.</li> <li>• Finally the consolidated marks sheet should be sent to the university and also to be made available at LIC visit.</li> </ul>
Field Visit, Plan, Discussion	10 Marks	
Commencement of activities and its progress	20 Marks	
Case study based Assessment Individual performance with report	20 Marks	
Sector wise study & its consolidation 5*5 = 25	25 Marks	
Video based seminar for 10 minutes by each student At the end of semester with Report. Activities 1 to 5, 5*5 = 25	25 Marks	
Total marks for the course in each semester	100 Marks	

For each activity, 20 marks CIE will be evaluated for IA marks at the end of semester, Report and assessment copy should be made available in the department.

Students should present the progress of the activities as per the schedule in the prescribed practical session in the field. There should be positive progress in the vertical order for the benefit of society in general through activities.

## INTRODUCTION TO ANDROID PROGRAMMING

<b>Course Code</b>	<b>24CDL381</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours /Week(L:T:P)</b>	<b>(0:0:2)</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>12 Lab slots</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>01</b>	<b>Exam Hours</b>	<b>03</b>

### Course objectives:

1. Outline the Android SDK features and the Development Framework and understanding Activities.
2. Learn adaptive, responsive user interfaces that work across a wide range of devices.
3. Identify background work and long-running tasks in Android applications
4. Describe the concepts of Storing, sharing and retrieving data in Android applications
5. Learn permissions, security and performance affect applications.

### Lab Experiments

1	Installation of Android studio.
2	Development Of Hello World Application.
3	Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.
4	Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button (use any layout).
5	Design an android application to create page using Intent and one Button and pass the Values from one Activity to second Activity.
6	Design an android application Send SMS using Internet.
7	Create an android application using Fragments.
8	Design an android application Using Radio buttons.
9	Design an android application for menu.
10	Create a user registration application that stores the user details in a database table.

### Course outcomes:

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

- Comprehend the basic features of Android Platform and Create Activities in Android.
- Demonstrate the design concepts of user interface using components and views in Android.
- Create and use databases for Android Application.
- Implement messaging services in Android.
- Deploy mobile applications in various market place for distribution

Sl. No	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
<b>Text book</b>				
1	Beginning Android Application Development	Wei – Meng Lee	Wiley publications	2007
2	Professional Android 4 Application Development	Alex Berson and Stephen J. Smith	Wiley publications	2012

### Reference Books

1	Beginning Android 3	Mark Murphy	Apress Springer India Pvt. Ltd	1st Edition, 2011
2	Pro Android 4	Sayed Hashimi, Satya Komatineni, Dave MacLean	Apress Springer India Pvt Ltd	1st Edition, 2012
3	Professional Android 2 Application Development	Reto Meier	Wiley India Pvt. Ltd	1st Edition, 2012

**Web links and Video Lectures:**

- 1) <https://developers.google.com/training/adf>
- 2) <https://goo.gl/ADKvq8>
- 3) <https://innovator.samsungmobile.com>

**Assessment Details(both IAT and SEE)**

<b>Continuous Internal Assessment of Laboratory/Practical Courses</b>		
Lab Test 1	Lab Test 2	Lab Records
15 marks	15 marks	20 marks
Semester End Examination(SEE)		50 marks

## ADVANCED PYTHON PROGRAMMING

<b>Course Code</b>	<b>24CDL382</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours /Week(L:T:P)</b>	<b>(0:0:2)</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>12 Lab slots</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>01</b>	<b>Exam Hours</b>	<b>03</b>

### Course objectives:

1. Learn Syntax and Semantics and create Functions in Python.
2. Handle Strings and Files in Python.
3. Understand Lists, Dictionaries and Regular expressions in Python.
4. Implement Object Oriented Programming concepts in Python.
5. Build Web Services and introduction to Network and Database Programming in Python.

### Lab Experiments

1	<p>Data Analysis and Visualization:</p> <ul style="list-style-type: none"> <li>• Implement a program to analyze datasets using Pandas, perform data cleaning, manipulation, and generate visualizations using Matplotlib or Seaborn.</li> </ul>
2	<p>Machine Learning Projects:</p> <ul style="list-style-type: none"> <li>• Develop a machine learning model to predict something relevant to the engineering field, such as predicting energy consumption, stock prices, or system failure using libraries like Scikit-learn or Tensor Flow.</li> </ul>
3	<p>Web Scraping and API Integration:</p> <ul style="list-style-type: none"> <li>• Create a program that scrapes data from websites or integrates with public APIs to gather engineering-related information. This could involve retrieving real-time weather data, stock market data, or scientific data from APIs.</li> </ul>
4	<p>Internet of Things (IoT) Projects:</p> <ul style="list-style-type: none"> <li>• Use Python to interface with IoT devices, sensors, or actuators. For example, create a program to read data from sensors and control IoT devices using platforms like Raspberry Pi or Arduino.</li> </ul>
5	<p>Network Analysis and Security:</p> <ul style="list-style-type: none"> <li>• Develop a program to analyze network traffic or simulate network protocols. Explore cybersecurity concepts by implementing basic encryption/decryption algorithms or network packet analysis tools.</li> </ul>
6	<p>Data Structures and Algorithms:</p> <ul style="list-style-type: none"> <li>• Implement advanced data structures (e.g., graphs, trees, heaps) and algorithms (e.g., sorting, searching, dynamic programming) in Python. Demonstrate their applications and efficiencies in solving engineering-related problems.</li> </ul>
7	<p>Computer Vision Projects:</p> <ul style="list-style-type: none"> <li>• Use libraries like OpenCV to work on image processing and computer vision tasks. Develop programs for object detection, face recognition, or gesture recognition.</li> </ul>
8	<p>Natural Language Processing (NLP) Projects:</p> <ul style="list-style-type: none"> <li>• Create programs that analyze engineering-related texts. This could involve sentiment analysis on engineering-related articles, technical document summarization, or keyword extraction from research papers.</li> </ul>
9	<p>Parallel and Concurrent Programming:</p> <ul style="list-style-type: none"> <li>• Explore the concepts of multithreading, multiprocessing, and asynchronous programming in Python. Develop programs that utilize these concepts to solve engineering problems, emphasizing performance improvements.</li> </ul>
10	<p>GUI Applications and User Interfaces:</p> <ul style="list-style-type: none"> <li>• Design and develop GUI applications using libraries like Tkinter or PyQt. Create engineering tools or simulations with user-friendly interfaces for tasks like circuit simulation, mathematical modeling, or data analysis.</li> </ul>

**Course Outcomes**

- CO1: Examine Python syntax and semantics and be fluent in the use of Python flow control and functions
- CO2: Demonstrate proficiency in handling Strings and File Systems
- CO3: Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.
- CO4: Interpret the concepts of Object-Oriented Programming as used in Python.
- CO5: Implement exemplary applications related to Network Programming, Web Services and Databases in Python.

**Text Books:**

1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.

**References:**

1. Think Python, Allen Downey, Green Tea Press
2. Introduction to Python, Kenneth A. Lambert, Cengage
3. Python Programming: A Modern Approach, VamsiKurama, Pearson

**Nptel Resources:**

1. NOC: Machine Learning, ML (Video): <https://nptel.ac.in/courses/106106202/>
2. NOC: Introduction to Machine Learning (Video): <https://nptel.ac.in/courses/106105152/>
3. NOC: Introduction to Machine Learning (Course sponsored by Aricent) (Video): <https://nptel.ac.in/courses/106106139/>

**Assessment Details(both IAT and SEE)**

<b>Continuous Internal Assessment of Laboratory/Practical Courses</b>		
Lab Test 1	Lab Test 2	Lab Records
15 marks	15 marks	20 marks
Semester End Examination(SEE)		50 marks

## INTRODUCTION TO OFFICE TOOLS

<b>Course Code</b>	<b>24CDL383</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours /Week(L:T:P)</b>	<b>(0:0:2)</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>12 Lab slots</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>01</b>	<b>Exam Hours</b>	<b>03</b>

### Course Objectives:

1. Attain a comprehensive understanding of Microsoft Office tools including Word, Excel, PowerPoint, and potentially others like Access, Outlook, or Publisher.
2. Learn how to use Office tools effectively to increase productivity in academic, professional, and personal tasks.
3. Develop the ability to adapt and apply acquired skills to different scenarios and tasks requiring Office applications.

### Lab Experiments

#### MS WORD:

1	Formatting tools like font styles, sizes, colors, alignment, and line spacing.
2	Create and customize document templates for different purposes (e.g., resumes, flyers, reports).
3	Explore creating and formatting tables, inserting and formatting images, shapes, and SmartArt.
4	Introduce basic arithmetic operations, SUM, AVERAGE, MAX, MIN, etc.

#### Microsoft PowerPoint:

5	Creating Presentations: Cover slide layouts, inserting text, images, shapes, and slide transitions.
6	Animation and Multimedia: Practice adding animations, audio, video, and customizing timing.
7	Master Slides: Explore using master slides for consistent formatting and design.
8	Microsoft Access: Design and create a simple database, define relationships, and run queries.
9	<b>Outlook:</b> Manage emails, calendar appointments, tasks, and rules for organizing mail.
10	<b>OneNote:</b> Explore note-taking, organizing information, and collaboration features.
11	<b>Publisher:</b> Designing various print materials like newsletters, brochures, or posters.
12	<b>SharePoint:</b> Introduction to document management, team sites, and collaboration tools.

### Course Outcomes (Course Skill Set):

- At the end of the course the student will be able to:
- CO 1. Know the basics of computers and prepare documents, spreadsheets, make small presentations with audio, video and graphs and would be acquainted with internet.
  - CO 2. Create, edit, save and print documents with list tables, header, footer, graphic, spellchecker, mail merge and grammar checker
  - CO 3. Attain the knowledge about spread sheet with formula, macros spell checker etc.
  - CO 4. Demonstrate the ability to apply application software in an office environment.
  - CO 5. Use Google Suite for office data management tasks

**Text Books:**

1. New Perspectives Microsoft Office 365 & Office 2019 Introductory  
Patrick Carey, Katherine T. Pinard, Ann Shaffer, Mark Shellman
2. Microsoft Office 2010 Introductory  
Gary B. Shelly, Misty E. Vermaat
3. Microsoft PowerPoint 2010  
Illustrated Brief - Office 2010)  
David W. Beskeen

**Digital Resources:**

1. <https://www.geeksforgeeks.org/what-are-office-tools/>
2. [https://www.tutorialspoint.com/basics\\_of\\_computers/basics\\_of\\_computers\\_office\\_tools.htm](https://www.tutorialspoint.com/basics_of_computers/basics_of_computers_office_tools.htm)
3. <https://www.coursera.org/courses?query=microsoft%20office>

**Assessment Details(both IAT and SEE)**

<b>Continuous Internal Assessment of Laboratory/Practical Courses</b>		
Lab Test 1	Lab Test 2	Lab Records
15 marks	15 marks	20 marks
Semester End Examination(SEE)		50 marks

## INTRODUCTION TO LINUX/UNIX SHELL PROGRAMMING

<b>Course Code</b>	<b>24CDL384</b>	<b>CIE Marks</b>	<b>50</b>
<b>Teaching Hours /Week(L:T:P)</b>	<b>(0:0:2)</b>	<b>SEE Marks</b>	<b>50</b>
<b>Total Hours of Pedagogy</b>	<b>-</b>	<b>Total Marks</b>	<b>100</b>
<b>Credits</b>	<b>01</b>	<b>Exam Hours</b>	<b>03</b>

### Course objectives:

This course will enable students to,

1. Understand effective use of Unix concepts, commands and terminology.
2. Identify, access, and evaluate UNIX file system.
3. Understand UNIX command syntax and semantics.
4. Read and understand specifications, scripts and programs.
5. Analyze Facility with UNIX Process.

### Course Content:

#### Introduction to Shell scripting:

- Use of Basic UNIX Shell Commands and options related to them: vi, ls, mkdir, rmdir, cd, cat, touch, file, wc, sort, cut, who, man etc.
- Commands related to inode, I/O redirection and piping.
- Shell Programming: Shell script exercises based on following:

- (i) Interactive shell scripts
- (ii) Positional parameters
- (iii) Arithmetic
- (iv) if-then-fi, if-then- else-fi, nested if-else
- (v) Logical operators
- (vi) else + if equals elif, case structure
- (vii) while, until, for loops, use of break

#### Programs/Assignment on:

1. Write a shell script to check whether the entered username and password is valid or not.
2. Write a shell script to add, subtract, multiply, divide two numbers and add two strings.
3. Write a shell script that accepts two file names as arguments, and checks the permissions of these files are similar or different.
4. Write a shell program to perform convert lowercase to uppercase using tr statement.
5. Write a non-recursive shell script that accepts any number of arguments and prints them in a reverse order.
6. Write a shell script to check the given file is a directory or not.
7. Write a shell script to compute GCD & LCM of two numbers.
8. Write a shell script to find whether a given number is prime.
9. Write a shell script to check whether the given year is Leap year or not.
10. Write a shell script to check whether the given string is Palindrome or not.

### Course Outcomes

- CO1: Know the basics of Unix concepts and commands.  
CO2: Evaluate the UNIX file system.  
CO3: Apply Changes in file system.  
CO4: Write scripts and programs.  
CO5: Analyse Facility with UNIX system process.

### Text Book:

- 1) Sumitabha Das: "UNIX – Concepts and Applications", Tata McGraw Hill, Noida, 4th Edition, 15th Reprint, 2011, ISBN-13: 978-0-07-063546-3.
- 2) Behrouz A. Forouzan and Richard F. Gilberg: "UNIX and Shell programming", Cengage Learning, India, 1st Edition, 2005, ISBN: 81-35-0325-9.
- 3) M G Venkatesh Murthy: "UNIX and Shell programming", Pearson Education, Delhi, 1st Edition, 2005, ISBN: 81-7758-745-5.

**E-Resources:**

- 1) <http://www.mhhe.com/das/uca>
- 2) [http://www.tutorialspoint.com/unix/unix\\_tutorials.pdf](http://www.tutorialspoint.com/unix/unix_tutorials.pdf).
- 3) <http://www.perldoc.perl.org/>

**Assessment Details(both IAT and SEE)**

<b>Continuous Internal Assessment of Laboratory/Practical Courses</b>		
Lab Test 1	Lab Test 2	Lab Records
15 marks	15 marks	20 marks
Semester End Examination(SEE)		50 marks

**Conduct of Practical Examination:****Experiment distribution :**

**For laboratories having only one part:** Students are allowed to pick one experiment from the lot with equal opportunity.

**For laboratories having PART A and PART B:** Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (Need to change in accordance with university regulations)
  - a) For laboratories having only one part → Procedure + Execution + Viva-Voce:  
 $15+70+15 = 100$  Marks
  - b) For laboratories having PART A and PART B
    - i. Part A – Procedure + Execution + Viva =  $6 + 28 + 6 = 40$  Marks
    - ii. Part B – Procedure + Execution + Viva =  $9 + 42 + 9 = 60$  Marks