**III Semester**

**BE in Information Science & Engineering**
**Scheme of Teaching III Semester**
**Outcome Based Education and Choice Based Credit System (CBCS)**
**Effective from the academic year Batch - 2025-29**

| Sl. No | Course | Course Code | Course Title | Teaching Department (TD)/ Board | Teaching Hours /Week | | | TW & SL | TH/S | Examination | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | L | T | P | | | CIE Marks | SEE Marks | Total Marks | |
| 1 | BSC | 25IS31 | Probability, Distributions and Statistics | Mathematics | 3 | 2 | 0 | 50 | 120 | 50 | 50 | 100 | 4 |
| 2 | PCC | 25IS32 | Operating System | ISE | 3 | 0 | 0 | 48 | 90 | 50 | 50 | 100 | 3 |
| 3 | PCC | 25IS33 | Data Structures and Applications | ISE | 3 | 0 | 0 | 48 | 90 | 50 | 50 | 100 | 3 |
| 4 | IPCC | 25IS34 | Digital Design & Computer Organization | ISE | 3 | 0 | 2 | 50 | 120 | 50 | 50 | 100 | 4 |
| 5 | PCCL | 25ISL35 | Data Structure & Applications lab | ISE | 0 | 0 | 2 | 02 | 30 | 50 | 50 | 100 | 1 |
| 6 | ESC/ PLC | 25IS36X | ESC/ETC/ PLC | ISE | 2 | 0 | 2 | 34 | 90 | 50 | 50 | 100 | 3 |
| 7 | AEC | 25ISL37X | Ability Enhancement Course - III | ISE dept/ IT Industry | 0 | 0 | 2 | 02 | 30 | 50 | 50 | 100 | 1 |
| 8 | NCMC | 25NS38/ 25PE38/ 25YO38 | NSS/ PE/ YOGA | NSS/YOGA/ PE Coordinator | 0 | 0 | 0 | 28 | 28 | 100 | - | 100 | PP |
| 9 | SDC | 25SE39 | Community project)Project Based learning ot societal projecr) | | 0 | 0 | 0 | 30 | 30 | 50 | 50 | 100 | 1 |
| **Total** | | | | | | | | | **628** | **500** | **400** | **900** | **20** |
| 10 | NCMC | 25MDIP310 | Mathematics course for Lateral Entry Students | Maths Dept | 1 | 0 | 0 | 14 | 28 | 100 | - | 100 | PP |
| 11 | SK | 25ISSK311 | Skill Lab-1: Web Programming | ISE | 0 | 0 | 2 | 02 | 30 | 50 | 50 | 100 | 1 |
| **Total** | | | | | | | | | **686** | **650** | **450** | **1100** | **20+1** |

**Note:**
L: Lecture , T: Tutorial, P: Practical
TW & SL: Term Work & Self Learning, TH/S: Total Hous/Semester

| Engineering Science Course (ESC/ETC/PLC)– III | | | |
|---|---|---|---|
| 25IS36A | Object Oriented Programming with Java | 25IS36B | Object Oriented Programming with C++ |

| Ability Enhancement Course – III | | | |
|---|---|---|---|
| 25ISL37A | Data Analytics using Excel | 25ISL37C | Prompt Engineering |
| 25ISL37B | Project Management with Git | | |

| Probability, Distributions and Statistics | | | |
|---|---|---|---|
| **Course Code** | 25IS31 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 3:2:0 | **SEE Marks** | 50 |
| **Credits** | 04 | **Total Marks** | 100 |
| **Contact Hours** | 56 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge of

- Basics of Probability and Statistics
- Understanding of Hypothesis
- Basic Mathematics

**Course Objectives:**

This course will enable the students to:

**CLO1:** Introduce the concept of random variables, probability distributions, specific discrete and continuous distributions with practical application in Computer Science Engineering and social life situations.

**CLO2:** Provide the principles of statistical inferences and the basics of hypothesis testing with emphasis on some commonly encountered hypotheses.

**CLO3:** Determine whether an input has a statistically significant effect on the system's response through ANOVA testing

**Teaching-Learning Process (General Instructions)**

Teachers can use following strategies to accelerate the attainment of the various course outcomes.

1. Chalk and Talk with Black Board
2. ICT based Teaching
3. Demonstration based Teaching.

**MODULE-1** **12 Hours**

**Modular Arithmetic:** Introduction of modular arithmetic and its applications in Computer Science and Engineering. Introduction to Congruences, Linear Congruences, The Remainder theorem, Solving Polynomials, Linear Diophantine Equation, System of Linear Congruences, Euler's Theorem, Wilson Theorem and Fermat's little theorem. Applications of Congruences-RSA algorithm.

**Text Book: 3 – Chapters 4, 5, 6, 7 and 9.**

**MODULE-2** **11 Hours**

**Statistics:** Principles of least squares, Curve fitting by the method of least squares in the form: $y = a +$ , $y = a + bx + cx^2$, and $y = ax^b$. Correlation, Coefficient of correlation, Lines of regression, Angle between regression lines, rank correlation.

**Text Book: 1 – Chapters 11 and 12.**

**Text Book: 2 – Chapters 13 and 14.**

**MODULE-3** **11 Hours**

**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).

**Text Book: 1 – Chapters 3, 4, 5, 6.**
**Text Book: 2 – Chapters 6 and 7.**

**MODULE-4** **11 Hours**

**Joint Probability Distributions:** 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.

**Text Book: 1 - Chapters 3, 4, 5, 6.**
**Text Book: 4 – Chapter 8.**

**MODULE-5** **11 Hours**

**Statistical Inference:** Sampling distribution, standard error. Levels of significance. Testing of hypothesis. Test of significances. Confidence limits, simple sampling of attributes. Sampling variables, central limit theorem and confidences limit for unknown mean.

Test of significance for large samples (mean, proportion). Comparison of large samples (mean, proportion).

Test of Significance for means of two small samples, students-'t' distribution, and Chi-square distribution as a test of goodness of fit.

**Text Book: 1 – Chapters 8 and 10.**
**Text Book: 2 – Chapters 8, 9, 10 and 12.**

**Course outcomes (Course Skill Set):**

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

**CO1: Apply** the concepts of modular arithmetic, congruence's, linear Diophantine equations, Euler's theorem, Fermat's theorem, Wilson's theorem, and the RSA cryptographic algorithm to solve engineering and computational problems. (PO – 1,2,3)

. **CO2: Apply** least squares curve fitting, correlation, regression analysis, and rank correlation techniques to analyse engineering data and establish relationships between variables. (PO – 1,2,3)

**CO3: Solve** engineering problems involving discrete and continuous random variables using probability distributions, mathematical expectation, variance, and standard probability models such as Binomial, Poisson, and Normal distributions. (PO – 1,2,3)

**CO4: Analyse** joint probability distributions and Markov chains by determining expectations, covariance, correlation, transition probabilities, stationary distributions, and absorbing states for stochastic systems. (PO – 1,2,3)

**CO5:** **Apply** statistical inference techniques including hypothesis testing, confidence intervals, large and small sample tests, Student's *t*-test, Chi-square test, and the Central Limit Theorem to draw valid conclusions from engineering data. (PO – 1,2,3)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- There are 25 marks for the CIE's Assignment component and 25 for the Internal Assessment Test component.
- Each test shall be conducted for 25 marks. The first test will be administered after 40-50% of the coverage of the syllabus, and the second test will be administered after 85-90% of the coverage of the syllabus. The average of the two tests shall be scaled down to 25 marks.
- Any two assignment methods mentioned in the regulations; if an assignment is project-based then only one assignment for the course shall be planned. The schedule for assignments shall be planned properly by the course teacher. The teacher should not conduct two assignments at the end of the semester if two assignments are planned. Each assignment shall be conducted for 25 marks. (If two assignments are conducted then the sum of the two assignments shall be scaled down to 25 marks)
- The final CIE marks of the course out of 50 will be the sum of the scale-down marks of tests and assignment/s marks.

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:**

Theory SEE will be conducted by the University as per the scheduled timetable, with common question papers for the course (duration 03 hours).

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks.

**Suggested Learning Resources:**

**Text Books:**

1. Ronald E. Walpole, Raymond H Myers, Sharon L Myers & Keying Ye "Probability and Statistics for Engineers and Scientists", Pearson Education, Ninth edition, 2012.

2. Murray R. Spiegel, Larry J. Stephens, Statistics, Schaum's outline series, Third edition.
3. Koshy, Thomas. *Elementary number theory with applications*. Academic press, 2nd Edition, 2009.
4. Seymour Lipschutz and Marc Lars Lipson, Probability, Schaum's Outline Series, Second edition, 2011.

**Reference Books:**

1. Erwin Kreyszig, "Advanced Engineering Mathematics", John Wiley & Sons, 9 th Edition, 2006.
2. B. S. Grewal "Higher Engineering Mathematics", Khanna publishers, 44 th Ed., 2021.
3. Irwin Miller & Marylees Miller, John E. Freund's "Mathematical Statistics with Applications" Pearson. Dorling Kindersley Pvt. Ltd. India, 8th edition, 2014.
4. S C Gupta and V K Kapoor, "Fundamentals of Mathematical Statistics", S Chand and Company, Latest edition.
5. P. G. Hoel, S. C. Port and C. J. Stone, "Introduction to Probability Theory", Universal Book Stall, (Reprint), 2003.
6. S. Ross, "A First Course in Probability", Pearson Education India, 6 th Ed., 2002.
7. David M Burton: "Elementary Number Theory" Mc Graw Hill, 7th Ed.,2017

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

- Probability and statistics: https://nptel.ac.in/courses/111105090
- Introduction to Probability and Statistics for Machine Learning: https://www.classcentral.com/course/codesignal-introduction-to-probability-and-statistics-for-machine-learning-358081
- Probability and Statistics: https://www.classcentral.com/course/probability-statistics-11755 Introduction to probability, statistics, and random processes: https://probabilitycourse.com/
- Introduction to Probability and Statistics: https://ocw.mit.edu/courses/18-05-introduction-to-probability-and-statistics-spring-2022/
- VTU e-Shikshana Program
- VTU EDUSAT Program

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

**Learning Activity – 1: Tutorial:** Practing problems (Two objective type assessments for 5 marks each, total 10 marks) or Conduction of CIA (Continuous Internal Assessment) for 10 marks.

**Learning Activity – 2: Assignments: A**ssigning two assignments each for 5 marks, total 10 marks.

**Learning Activity – 3:** Choose either Seminar or Project based learning activity with report for 5 marks.

| OPERATING SYSTEMS | | | |
|---|---|---|---|
| **Course Code** | 25IS32 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L:T:P)** | 3:0:0 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 42 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisite:**

The Students Should have knowledge of
Computer Architecture / Organization

**Course objectives:**

**CLO1:** To provide students with a fundamental understanding of the structure, functions, and services of an operating system.

**CLO2:** To enable students to understand and apply CPU scheduling algorithms for efficient process management.

**CLO3:** To develop the ability to analyze process synchronization mechanisms and deadlock handling techniques used in operating systems.

**CLO4:** To familiarize students with memory management strategies and virtual memory concepts, and enable them to apply paging and page replacement techniques.

**CLO5:** To introduce file system organization, secondary storage management, and protection mechanisms, and develop the ability to apply disk scheduling techniques for efficient storage management.

**Teaching-Learning Process (General Instructions):**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**MODULE-1** **09 Hours**

Introduction to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Special-purpose systems; Computing environments.

Operating System Services: User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; Operating System debugging, Operating System generation; System boot.

**Textbook 1:**Ch 1.1 to 1.8, 2.1 to 2.11

**MODULE-2** **09 Hours**

Process Management: Process concept; Process scheduling; Operations on processes; Inter process communication

Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues.

Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Thread scheduling; Multiple-processor scheduling,

**Textbook 1:** Chapter – 3 (3.1-3.4), 4 (4.1-4.4), 5 (5.1 -5.5)

| MODULE-3 | 08 Hours |
|---|---|
| Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization;<br>Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.<br>**Textbook 1:** Chapter – 6 (6.1-6.6), 7 (7.1 -7.7) | |
| **MODULE-4** | **08 Hours** |
| Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation.<br>Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.<br>**Textbook 1:** Chapter -8 (8.1-8.6), 9 (9.1-9.6) | |
| **MODULE-5** | **08 Hours** |
| File System: File system: File concept; Access methods; Directory and Disk structure<br>Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management.<br>Secondary Storage Structure: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management;<br>Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix<br>**Textbook 1:** Chapter -10(10.1-10.3),11(11.1-11.5),12(12.1-12.5),14(14.1-14.4) | |

**Course outcome (Course Skill Set):**

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

**CO1:** Explain the structure and functionality of operating system. (PO-1,2,3,11, PSO-1,2)

**CO2:** Apply appropriate CPU scheduling algorithms for the given problem. (PO-1,2,3,11, PSO-1,2)

**CO3:** Analyse the various techniques for process synchronization and deadlock handling. (PO-1,2,3,11, PSO-1,2)

**CO4:** Explain memory management strategies and virtual memory concepts, and apply paging and page replacement techniques for efficient memory management. (PO-1,2,3,11, PSO-1,2)

**CO5:** Explain file system organization and protection mechanisms, and apply disk scheduling algorithms for efficient disk access. (PO-1,2,3,11, PSO-1,2)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation(CIE):**

● For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.

● The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

● Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.

● For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:** Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

**Suggested Learning Resources:**

**Textbooks**

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 8th edition, Wiley-India, 2015

**Reference Books**

**1**. Ann McHoes Ida M Fylnn, Understanding Operating System, Cengage Learning, 6th Edition
2. D.M Dhamdhere, Operating Systems: A Concept Based Approach 3rd Ed, McGraw- Hill, 2013.
3. P.C.P. Bhatt, An Introduction to Operating Systems: Concepts and Practice 4th Edition, PHI(EEE),2014.
4. William Stallings Operating Systems: Internals and Design Principles, 6th Edition, Pearson.

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

- https://youtu.be/mXw9ruZaxzQ 14.09.2023 MKV-TEMPLATE for IPCC (26.04.2022) Annexure-III
- https://youtu.be/vBURTt97EkA
- https://www.youtube.com/watch?v=783KABtuE4&list=PLIemF3uozcAKTgsCIj82voMK3TMR0YE_f
- https://www.youtube.com/watch?v=3-ITLMMeeXY&list=PL3pGy4HtqwD0n7bQfHjPnsWzkeRn6mkO

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

- Case Study
- Assignments
- Seminars
- MOOC Certifications

| DATA STRUCTURES AND APPLICATIONS | | | |
|---|---|---|---|
| **Course Code** | 25IS33 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T:P)** | 3:0:0 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 42 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge of;

- Basic Programming Skills
- Understanding of Algorithms
- Basic Mathematics
- Problem-Solving Skills

**Course Objectives:**

**CLO1:**To explain fundamentals of data structures and their applications.
**CLO2:**To illustrate representation of Different data structures such as Stack, Queues.
**CLO3:**To Design and Develop Solutions to problems using Linked Lists
**CLO4:**To discuss applications of Nonlinear Data Structures in problem solving such as trees
**CLO5:**To introduce advanced Data structure concepts such as Graphs and H a s h

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Module-1** **9 Hours**

**INTRODUCTION TO DATA STRUCTURES:** Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations pointers and dynamic Memory Allocation
**ARRAYS and STRUCTURES**: Arrays, Dynamic Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices, representation of Multidimensional Arrays, Strings
**Text Book**: Chapter-1:1.2 Chapter-2: 2.1 to 2.7

**Module-2** **8 Hours**

**Stacks and Queues**: Stacks, Stacks Using Dynamic Arrays, Queues, Circular QueuesUsing Dynamic Arrays, Evaluation of Expressions, Multiple Stacks and Queues.
**Text Book**: Chapter-3: 3.1,3.2,3.6 3.3, 3.4, 3.7

**Module-3** **8 Hours**

**Linked Lists**: Singly Linked lists and Chains, Representing Chains in C, LinkedStacks and Queues, Polynomials, Additional List operations, Sparse Matrices, Doubly Linked Lists.
**Text Book**: Chapter-4: 4.1 to 4.4 4.5,4.7,4.8

**Module-4** **8 Hours**

**TREES:** Introduction to trees, Binary Trees, Binary Tree Traversals, Threaded Binary Trees. Binary Search trees, Selection Trees, Forests, Representation of Disjoint sets, Counting Binary Trees, Leftist Trees. Efficient Binary Search Trees: AVL Trees, Red Black Tree
**Text Book**: Chapter-5: 5.1 to 5.3, 5.5 ,5.7 to 5.11

**Module-5** **9 Hours**

**GRAPHS:** The Graph Abstract Data Types, Elementary Graph Operations (BFS/DFS)
**HASHING:** Introduction, Static Hashing, Dynamic Hashing
**Text Book** Chapter-6: 6.1, 6.2 Chapter 8: 8.1 to 8.3

**Course outcomes (Course Skill Set):**

At the end of the course, the student will be able to:
**CO1:** Explain different data structures and their applications. (PO – 1,2,3,5, PSO – 1,3)
**CO2:** Apply Arrays, Stacks and Queue data structures to solve the given problems. (PO – 1,2,3,5, PSO – 1,3)
**CO3:** Use the concept of linked list in problem solving.(PO – 1,2,3,5, PSO – 1,3)
**CO4:**Develop solutions using trees and graphs to model the real-world problem.(PO – 1,2,3,5, PSO – 1,3)
**CO5:** Explain the advanced Data Structures concepts such as Hashing Techniques and Optimal Binary Search Trees.(PO – 1,2,3,5, PSO – 1,3)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation(CIE):**

● For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.

● The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

● Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.

● For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:** Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

**Suggested Learning Resources:**

**Textbook:**

1. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014

**Reference Books:**

1. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
2. Gilberg & Forouzan, Data Structures: A Pseudo-codeapproachwithC,2ndEd, Cengage Learning,2014.
3. Reema Thareja, Data Structures usingC,3rd Ed, Oxfordpress,2012.
4. Jean-Paul Tremblay &Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Ed, McGraw Hill, 2013.
5. A M Tenenbaum, Data Structures using C, PHI,1989
6. Robert Kruse, Data Structures and ProgramDesigninC,2ndEd, PHI,1996**.**

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

- http://elearning.vtu.ac.in/econtent/courses/video/CSE/06CS35.html
- https://nptel.ac.in/courses/106/105/106105171/
- http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html
- https://www.youtube.com/watch?v=3Xo6P_V-qns&t=201s
- https://ds2-iiith.vlabs.ac.in/exp/selection-sort/index.html
- https://nptel.ac.in/courses/106/102/106102064/
- https://ds1-iiith.vlabs.ac.in/exp/stacks-queues/index.html
- https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html
- https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html
- https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/index.html
- https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/depth-first-traversal/dft-practice.html
- https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013501595428077568125 59/overview

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

- Role Play
- Flipped classroom
- AssessmentMethodsfor25 Marks (opt two Learning Activities)
    - Case Study
    - Programming Assignment
    - Gate Based Aptitude Test
    - MOOC Assignment for selected Module

| Digital Design and Computer Organization | | | |
|---|---|---|---|
| **Course Code** | 25IS34 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 3:0:2 | **SEE Marks** | 50 |
| **Credits** | 04 | **Total Marks** | 100 |
| **Contact Hours** | 42 Theory + 28Practical | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge of

- Basic Electronics
- Digital Circuits
- Computer Architecture Concepts

**Course Objectives:**

CLO1:To demonstrate the functionalities of binary logic system
CLO2:To explain the working of combinational and sequential logic system
CLO3:To realize the basic structure of computer system
CLO4:To illustrate the working of I/O operations and processing unit

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Module-1** **9 Hours**

**Introduction to Digital Design:** Binary Logic, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Digital Logic Gates, Introduction, The Map Method, Four-Variable Map, Don't- Care Conditions, NAND and NOR Implementation, Other Hardware Description Language – Verilog Model of a simple circuit.

**Case Study:** Design a solution for a Real Time Problem Using K-Maps.

**Text book 1: 1.9, 2.4, 2.5, 2.8, 3.1, 3.2, 3.3, 3.5, 3.6, 3.9**

**Module-2** **9 Hours**

**Combinational Logic**: Introduction, Combinational Circuits, Design Procedure, Binary Adder-Subtractor, Decoders, Encoders, Multiplexers. HDL Models of Combinational Circuits – Adder, Multiplexer, Encoder.
**Sequential Logic**: Introduction, Sequential Circuits, Storage Elements: Latches, Flip-Flops.
**Case Study:** Design a real-world control system based on Combinational and Sequential logic circuits to manage specific operational requirements effectively.
**Text book 1: 4.1, 4.2, 4.4, 4.5, 4.9, 4.10, 4.11, 4.12, 5.1, 5.2, 5.3, 5.4.**

**Module-3** **8 Hours**

**Basic Structure of Computers:** Functional Units, Basic Operational Concepts, Bus structure, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instruction and Instruction sequencing, Addressing Modes.

**Text book 2**: 1.2, 1.3, 1.4, 1.6, 2.2, 2.3, 2.4, 2.5

**Module-4** **8 Hours**

**Input/output Organization:** Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access: Bus Arbitration, Speed, size and Cost of memory systems. Cache Memories – Mapping Functions.

**Case Study**: Optimizing Cache Mapping in an AI-Based System

**Text book 2:** 4.1, 4.2.1, 4.2.2, 4.2.3, 4.4, 5.4, 5.5.1

**Module-5** **8 Hours**

**Basic Processing Unit:** Some Fundamental Concepts: Register Transfers, Performing ALU operations, fetching a word from Memory, Storing a word in memory. Execution of a Complete Instruction. **Pipelining:** Basic concepts, Role of Cache memory, Pipeline Performance.

**Case Study:** Pipelining in AI-Powered

**Text book 2:** 7.1, 7.2, 8.1

**PRACTICAL COMPONENT OF IPCC**

| Sl. No. | Experiments<br>Simulation packages preferred: Multisim, Model sim, PSpice or any other relevant |
|---|---|
| 1. | Given a 4-variable logic expression, simulate the same using basic gates.<br>simplify it using appropriate technique and |
| 2. | Implement and simulate a 4-bit Full Adder using basic logic gates. |
| 3. | Implement and simulate a 4-bit Full Subtractor using basic logic gates. |
| 4. | Develop simple digital circuits using Verilog HDL based on Structural, Dataflow, and Behavioral modeling. |
| 5. | Design Verilog HDL to implement Binary Half Adder and Half Subtractor. |

| | |
|---|---|
| 6. | Design Verilog HDL to implement Binary Full Adder, and Full Subtractor. |
| 7. | Develop a Decimal Adder using Verilog HDL. |
| 8. | Design Verilog program to implement Different types of multiplexers like 2:1, 4:1 and 8:1. |
| 9. | Design Verilog program to implement types of De-Multiplexer. |
| 10. | Implement various types of Flip-Flops, such as SR, JK, and D Flip-Flops, using Verilog HDL. |

**Course outcomes (Course Skill Set):**

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

**CO1:** Apply the K–Map techniques to simplify various Boolean expressions. (PO – 1,2,3 PSO – 1,3)

**CO2:** Design different types of combinational and sequential circuits along with Verilog programs. (PO – 1,2,3,5, PSO – 1,3)

**CO3:** Describe the fundamentals of machine instructions, addressing modes and Processor performance. (PO – 1,2,3,5 PSO – 1)

**CO4:** Explain the approaches involved in achieving communication between processor and I/O devices. (PO – 1,2,3, PSO – 1)

**CO5:** Analyze internal Organization of Memory and Impact of cache/Pipelining on Processor Performance. (PO – 1,2,3, PSO – 1)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**CIE for the theory component of the IPCC (maximum marks 50)**

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are 25 marks and that for the practical component is 25 marks.
- 25 marks for the theory component are split into 15 marks for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and 10 marks for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.

Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for 25 marks).

- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**CIE for the practical component of the IPCC**

1. 15 marks for the conduction of the experiment and preparation of laboratory record, and 10 marks for the test to be conducted after the completion of all the laboratory sessions.
2. On completion of every experiment/program in the laboratory, the students shall be evaluated including viva- voce and marks shall be awarded on the same day.
3. The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks of all experiments' write- ups are added and scaled down to 15 marks.
4. The laboratory test (duration 02/03 hours) after completion of all the experiments shall be conducted for 50 marks and scaled down to 10 marks.
5. Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for 25 marks.
6. The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

**Semester-End Examination:**

Theory SEE will be conducted by the University as per the scheduled timetable, with common question papers for the course (duration 03 hours).

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks.

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.

**Suggested Learning Resources:**

**Textbook:**

1. M. Morris Mano & Michael D. Ciletti, Digital Design with an Introduction to Verilog Design, 5e, Pearson Education.

2. Carl Hamacher, Zvonko Vranesic, SafwatZaky, Computer Organization, 5th Edition, Tata McGraw Hill.

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

- https://elearning.vtu.ac.in/econtent/courses/web/CSE/15CS44.html
- https://cse11-iiith.vlabs.ac.in/
- https://infyspringboard.onwingspan.com
- https://onlinecourses.nptel.ac.in/noc21_ee39/preview

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

- Role Play
- Flipped classroom

**Assessment Methods for 25 Marks (opt two Learning Activities)**

- Case Study
- Programming Assignment

| DATA STRUCTURE AND APPLICATIONS LAB | | | |
|---|---|---|---|
| **Course Code** | 25ISL35 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 0:0:2 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 28 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Practical | | |

**Prerequisites:**

The students should have knowledge on

- Basic Programming Skills
- Understanding of Algorithms
- Basic Mathematics
- Problem-Solving Skills

**Course Objectives:**

**CLO1:**This laboratory courses enables students to get practical experience in design, develop, implement, analyze and evaluation/testing of Dynamic memory management
**CLO2:**Linear data structures and their applications such as stacks, queues and lists
**CLO3:**Non-Linear data structures and their applications such as trees and graphs

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Programs List:**

| | |
|---|---|
| 1. | Develop a Program in C for the following:<br>Declare a calendar as an array of 7 elements (A dynamically Created array) to represent 7 days of a week. Each Element of the array is a structure having three fields. The first field is the name of the Day (A dynamically allocated String), The second field is the date of the Day (A integer), the third field is the description of the activity for a particular day (A dynamically allocated String).<br><br>Write functions create(), read() and display(); to create the calendar, to read the data from the keyboard and to print weeks activity details report on screen. |
| 2. | Develop a Program in C for the following operations on Strings.<br>a. Read a main String (STR), a Pattern String (PAT) and a Replace String (REP)<br>Perform Pattern Matching Operation: Find and Replace all occurrences of PAT in STR with REP if PAT exists in STR. Report suitable messages in case PAT does not exist in STR<br><br>Support the program with functions for each of the above operations. Don't use Built-in functions. |

| | |
|---|---|
| 3. | Develop a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX)<br>Push an Element on to Stack<br>Pop an Element from Stack<br>Demonstrate how Stack can be used to check Palindrome<br>Demonstrate Overflow and Underflow situations on Stack<br>Display the status of Stack<br>Exit<br>Support the program with appropriate functions for each of the above operations |
| 4 | Develop a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, % (Remainder), ^ (Power) and alphanumeric operands. |
| 5. | Develop a Program in C for the following Stack Applications<br>a. Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^<br>b. Solving Tower of Hanoi problem with n disks |
| 6. | Develop a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX)<br>Insert an Element on to Circular QUEUE<br>Delete an Element from Circular QUEUE<br>Demonstrate Overflow and Underflow situations on Circular QUEUE<br>Display the status of Circular QUEUE<br>Exit<br><br>Support the program with appropriate functions for each of the above operations |
| 7 | Develop a menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: *USN, Name, Programme, Sem, PhNo*<br>Create a SLL of N Students Data by using *front insertion.*<br>Display the status of SLL and count the number of nodes in it<br>Perform Insertion / Deletion at End of SLL<br>Perform Insertion / Deletion at Front of SLL(Demonstration of stack) |
| 8 | Develop a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: *SSN, Name, Dept, Designation, Sal, PhNo*<br>Create a DLL of N Employees Data by using *end insertion.*<br>Display the status of DLL and count the number of nodes in it<br>Perform Insertion and Deletion at End of DLL<br>Perform Insertion and Deletion at Front of DLL<br>Demonstrate how this DLL can be used as Double Ended Queue. |
| 9 | Develop a Program in C for the following operations on Singly Circular Linked List(SCLL) with header nodes<br>. Represent and Evaluate a Polynomial $P(x, y) = 6x^2 y^2 -4y +3x^3 y+2xy^5 -2xy$<br><br>Find the sum of two polynomials POLY1(x, y) and POLY2(x, y) and store the result in POLYSUM(x, y) |

| | |
|---|---|
| 10 | Develop a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers.<br>Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2<br>Traverse the BST in Inorder, Preorder and Post Order<br>Search the BST for a given element (KEY) and report the appropriate message |
| 11 | Develop a Program in C for the following operations on Graph(G) of Cities<br>Create a Graph of N cities using Adjacency Matrix.<br><br>Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method |
| 12. | Given a File of N employee records with a set K of Keys (4-digit) which uniquely determine the records in file F. Assume that file F is maintained in memory by a Hash Table (HT) of m memory locations with L as the set of memory addresses (2-digit) of locations in HT. Let the keys in K and addresses in L are Integers. Develop a Program in C that uses Hash function H: K →L as H(K)=K mod m (remainder method), and implement hashing technique to map a given key K to the address space L. Resolve the collision (if any) using linear probing. |

**Course outcomes (Course Skill Set):**

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

**CO1:** Analyze various linear and non-linear data structures (PO1,2,3, PSO-1,3)

**CO2:** Demonstrate the working nature of different types of data structures and their applications (PO1,2,3, PSO-1,3)

**CO3:** Use appropriate searching and sorting algorithms for the give scenario. (PO1,2,3, PSO-1,3)

**CO4:** Apply the appropriate data structure for solving real world problems(PO1,2,3, PSO-1,3)

**CO5:**Develop solution to solve real world problems.

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- CIE marks for the practical course are 50 Marks. The split-up of CIE marks for record/ journal and test are in the ratio 60:40.
- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to 20 marks (40% of the maximum marks). The Sum of scaled-down

marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student

• Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**Semester-End Examination:**

SEE marks for the practical course are 50 Marks.

● SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.

● The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University

All laboratory experiments are to be included for practical examination.

● (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.

● Students can pick one question from PART-A and one question from PART-B (experiment) from the questions lot prepared by the examiners jointly and 50% weightage for each part.

● Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

● General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners) Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero. The minimum duration of SEE is 02 hours

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

- https://ds1-iiith.vlabs.ac.in/exp/stacks-queues/index.html
- https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html
- https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html
- https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/index.html
- https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/depth-first-traversal/dft-practice.html

| OBJECTED ORIENTED PROGRAMMING WITH JAVA | | | |
|---|---|---|---|
| **Course Code** | 25IS36A | **CIE Marks** | 50 |
| **Teaching Hours/Week (L:T:P)** | 2:0:2 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 28 Theory +28 Practical | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisite:**

The students should have knowledge of

- Basic Programming Skills:
- Knowledge of C & C++

**Course objectives:**

**CLO1:**To learn primitive constructs JAVA programming language.
**CLO2:**To understand Object Oriented Programming Features of JAVA.
**CLO3:**To gain knowledge on: packages, multithreaded programing and exceptions

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Module-1** **6 Hours**

**An Overview of Java**: Object-Oriented Programming (Two Paradigms, Abstraction, The Three OOP Principles), Using Blocks of Code, Lexical Issues (Whitespace, Identifiers, Literals, Comments, Separators, The Java Keywords).

**Data Types, Variables, and Arrays**: The Primitive Types (Integers, Floating-Point Types, Characters, Booleans), Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays, Introducing Type Inference with Local Variables.

**Operators:** Arithmetic Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses.

**Control Statements**: Java's Selection Statements (if, The Traditional switch), Iteration Statements (while, do-while, for, The For-Each Version of the for Loop, Local Variable Type Inference in a for Loop, Nested Loops), Jump Statements (Using break, Using continue, return).

Textbook: Chapter 2, 3, 4, 5

**Module-2** **6 Hours**

**Introducing Classes:** Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection.

**Methods and Classes:** Overloading Methods, Objects as Parameters, Argument Passing, Returning Objects, Recursion, Access Control, Understanding static, Introducing final, Introducing Nested and Inner Classes.

Textbook: Chapter 6, 7

**Module-3** **6 Hours**

**Inheritance:** Inheritance Basics, Using super, Creating a Multilevel Hierarchy, When Constructors Are Executed, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, Local Variable Type Inference and Inheritance, The Object Class.

**Interfaces:** Interfaces, Default Interface Methods, Use static Methods in an Interface, Private Interface Methods.

Textbook: Chapter 8, 9

**Module-4** **5 Hours**

**Packages:** Packages, Packages and Member Access, Importing Packages.

**Exceptions:** Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions.

Textbook: Chapter 9, 10

**Module-5** **5 Hours**

**Multithreaded Programming:** The Java Thread Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Interthread Communication, Suspending, Resuming, and Stopping Threads, Obtaining a Thread's State.

**Enumerations, Type Wrappers and Autoboxing:** Enumerations (Enumeration Fundamentals, The values() and valueOf() Methods), Type Wrappers (Character, Boolean, The Numeric Type Wrappers), Autoboxing (Autoboxing and Methods, Autoboxing/Unboxing Occurs in Expressions, Autoboxing/Unboxing Boolean and Character Values).

Textbook: Chapter 11, 12

**PRACTICAL COMPONENT**

1. Develop a JAVA program to add TWO matrices of suitable order N (The value of N should be read from command line arguments).
2. Develop a stack class to hold a maximum of 10 integers with suitable methods. Develop a JAVA main method to illustrate Stack operations.
3. A class called Employee, which models an employee with an ID, name and salary, is designed as shown in the following class diagram. The method raiseSalary (percent) increases the salary by the given percentage. Develop the Employee class and suitable main method for demonstration.

4. A class called MyPoint, which models a 2D point with x and y coordinates, is designed as follows:
   - Two instance variables x (int) and y (int).
   - A default (or "no-arg") constructor that construct a point at the default location of (0, 0).

- A overloaded constructor that constructs a point with the given x and y coordinates.
- A method setXY() to set both x and y.
- A method getXY() which returns the x and y in a 2-element int array.
- A toString() method that returns a string description of the instance in the format "(x, y)".
- A method called distance(int x, int y) that returns the distance from this point to another point at the given (x, y) coordinates
- An overloaded distance(MyPoint another) that returns the distance from this point to the given MyPoint instance (called another)
- Another overloaded distance() method that returns the distance from this point to the origin (0,0)

Develop the code for the class MyPoint. Also develop a JAVA program (called TestMyPoint) to test all the methods defined in the class.

5. Develop a JAVA program to create a class named shape. Create three sub classes namely: circle, triangle and square, each class has two member functions named draw () and erase (). Demonstrate polymorphism concepts by developing suitable methods, defining member data and main program.

6. Develop a JAVA program to create an abstract class Shape with abstract methods calculateArea() and calculate Perimeter(). Create subclasses Circle and Triangle that extend the Shape class and implement the respective methods to calculate the area and perimeter of each shape.

7. Develop a JAVA program to create an interface Resizable with methods resizeWidth(int width) and resize Height(int height) that allow an object to be resized. Create a class Rectangle that implements the Resizable interface and implements the resize methods

8. Develop a JAVA program to create an outer class with a function display. Create another class inside the outer class named inner with a function called display and call the two functions in the main class.

9. Develop a JAVA program to raise a custom exception (user defined exception) for DivisionByZero using try, catch, throw and finally.

10. Develop a JAVA program to create a package named mypack and import & implement it in a suitable class.

11. Write a program to illustrate creation of threads using runnable class. (start method start each of the newly created thread. Inside the run method there is sleep() for suspend the thread for 500 milliseconds).

12. Develop a program to create a class MyThread in this class a constructor, call the base class constructor, using super and start the thread. The run method of the class starts after this. It can be observed that both main thread and created child thread are executed concurrently.

**Course outcome (Course Skill Set)**

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

**CO1:** Demonstrate proficiency in writing simple programs involving branching and looping structures. (PO-1,2,3,5 PSO-1,3)

**CO2:** Design a class involving data members and methods for the given scenario. (PO-1,2,3,5, PSO-1,2,3)

**CO3:** Apply the concepts of inheritance and interfaces in solving real world problems. (PO-1,2,3,5, PSO-1,2,3)

**CO4:** Use the concept of packages and exception handling in solving complex problem (PO-1,2,3,5, PSO-1,2,3)

**CO5:** Apply concepts of multithreading, autoboxing and enumerations in program development(PO-1,2,3,5, PSO-1,2,3)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to

have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**CIE for the theory component of the IPCC (maximum marks 50)**

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.
- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for 25 marks).
- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC.

**CIE for the practical component of the IPCC**

- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report.
- 15 marks for the conduction of the experiment and preparation of laboratory record, and 10 marks for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to 15 marks.
- The laboratory test after completion of all the experiments shall be conducted for 50 marks and scaled down to 10 marks.
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for 25 marks.
- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

**SEE for IPCC**

Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course (duration 03 hours)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

**The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.**

**Suggested Learning Resources**

**Text Books:**

**1.** Java: The Complete Reference, Twelfth Edition, by Herbert Schildt, November 2021, McGraw-Hill, ISBN: 9781260463422

**Reference Books:**

1.Programming with Java, 6th Edition, by E Balagurusamy, Mar-2019, McGraw Hill Education, ISBN:9789353162337.

2.Thinking in Java, Fourth Edition, by Bruce Eckel, Prentice Hall, 2006 (https://sd.blackball.lv/library/thinking_in_java_4th_edition.pdf)

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

- 1.Basics of C++ - https://www.youtube.com/watch?v=BClS40yzssA
- Functions of C++ - https://www.youtube.com/watch?v=p8ehAjZWjPw

**Tutorial Link:**

- https://www.w3schools.com/cpp/cpp_intro.asp
- https://www.edx.org/course/introduction-to-c-3
- 3.https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013843642506788864433 75_shared/overview

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

- Flipped classroom
- Assessment Methods for 25 Marks (opt two Learning Activities)
  - Case Study
  - Programming Assignment
  - MOOC Assignment for selected Module

| OBJECT ORIENTED PROGRAMMING WITH C++ | | | |
|---|---|---|---|
| **Course Code** | 25IS36B | **CIE Marks** | 50 |
| **Teaching Hours/Week (L:T:P)** | 2:0:2 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 28 Theory + 28 Practical | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisite:**

The students should have knowledge of

- Basic Programming Skills:
- Knowledge of C

**Course objectives:**

**CLO1:**To understand object-oriented programming using C++and Gain knowledge about the capability to store information together in an object.
**CLO2:**To illustrate the capability of a class to rely upon another class and functions.
**CLO3:**To Create and process data in files using file I/O functions
**CLO4:**To understand the generic programming features of C++ including Exception handling

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Module-1** **6 Hours**

An overview of C++: What is object-Oriented Programming? Introducing C++ Classes, The General Form of a C++ Program. Classes and Objects: Classes, Friend Functions, Friend Classes, Inline Functions, Parameterized Constructors, Static Class Members, When Constructors and Destructors are Executed, The Scope Resolution Operator, Passing Objects to functions, Returning Objects, Object Assignment
Text book: Ch 11, Ch 12

**Module-2** **6 Hours**

Arrays, Pointers, References, and the Dynamic Allocation Operators: Arrays of Objects, Pointers to Objects, The this Pointer, Pointers to derived types, Pointers to class members. Functions Overloading, Copy Constructors: Functions Overloading, Overloading Constructor Functions. Copy Constructors, Default Function Arguments, Function Overloading and Ambiguity.
Text book: Ch 13, Ch 14

**Module-3** **5 Hours**

Operator Overloading: Creating a Member Operator Function, Operator Overloading Using a Friend Function, Overloading new and delete Inheritance: Base-Class Access Control, Inheritance and Protected

Members, Inheriting Multiple Base Classes, Constructors, Destructors and Inheritance, Granting Access, Virtual Base Classes

Text book: Ch 15, Ch 16

**Module-4** **6 Hours**

Virtual Functions and Polymorphism: Virtual Functions, The Virtual Attribute is Inherited, Virtual Functions are Hierarchical, Pure Virtual Functions, Using Virtual Functions, Early vs Late Binding. Templates: Generic Functions, Applying Generic Functions, Generic Classes. The type name and export Keywords. The Power of Templates

Text book: Ch 17, Ch 18

**Module-5** **5 Hours**

Exception Handling: Exception Handling Fundamentals, Handling Derived-Class Exceptions, Exception Handling Options, Applying Exception Handling. The C++ I/O System Basics: C++ Streams, The C++ Classes, Formatted I/O File I/O: and File Classes, Opening and Closing a File, Reading and Writing Text Files, Detecting EOF.

Text book: Ch 19, Ch 20, Ch21

**PRACTICAL COMPONENT**

| Si. NO | EXPERIMENT NAME |
|---|---|
| **1** | Develop a C++ program to find the largest of three numbers |
| **2** | Develop a C++ program to sort the elements in ascending and descending order. |
| **3** | Develop a C++ program using classes to display student name, roll number, marks obtained in two subjects and total score of student |
| **4** | Develop a C++ program for a bank empolyee to print name of the employee, account_no. & balance. Print invalid balance if amount<500, Display the same, also display the balance after withdraw and deposit. |
| **5** | Develop a C++ program to demonstrate function overloading for the following prototypes. add(int a, int b), add(double a, double b |
| **6** | Develop a C++ program using Operator Overloading for overloading Unary minus operator. |
| **7** | Develop a C++ program to implement Multiple inheritance for performing arithmetic operation of two numbers |
| **8** | Develop a C++ program using Constructor in Derived classes to initialize alpha, beta and gamma and display corresponding values. |
| **9** | Develop 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. |
| **10** | Develop a C++ program to write and read time in/from binary file using fstream |
| **11** | Develop a function which throws a division by zero exception and catch it in catch block. Write a C++ program to demonstrate usage of try, catch and throw to handle exception. |
| **12** | Develop a C++ program that handles array out of bounds exception using C++. |

**Course outcome (Course Skill Set)**

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

**CO1:** Illustrate the basic concepts of object-oriented programming.(PO-1,2,5,11, PSO-1,2,3)
**CO2:** Design appropriate classes for the given real world scenario. (PO-1,2,5,11, PSO-1,2,3)
**CO3:** Apply the knowledge of compile-time / run-time polymorphism to solve the given problem. (PO-1,2,5,11, PSO-1,2,3)
**CO4:** Use the knowledge of inheritance for developing optimized solutions(PO-1,2,5,11, PSO-1,2,3)
**CO5:** Apply the concepts of templates and exception handling for the given problem(PO-1,2,5,11, PSO-1,2,3)
**CO6:** Use the concepts of input output streams for file operations (PO-1,2,5,11, PSO-1,2,3)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**CIE for the theory component of the IPCC (maximum marks 50)**

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.
- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for 25 marks).
- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC.

**CIE for the practical component of the IPCC**

- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report.
- 15 marks for the conduction of the experiment and preparation of laboratory record, and 10 marks for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to 15 marks.
- The laboratory test after completion of all the experiments shall be conducted for 50 marks and scaled down to 10 marks.
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for 25 marks.
- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

**SEE for IPCC**

Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course

(duration 03 hours)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

**The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.**

**Suggested Learning Resources:**

**Textbooks**


1. Herbert schildt, The Complete Reference C++, 4th edition, TMH, 2005


**Reference Books**


1. Balagurusamy E, Object Oriented Programming with C++, Tata McGraw Hill
2. Education Pvt.Ltd., Sixth Edition 2016.
3. Bhave , " Object Oriented Programming With C++", Pearson Education , 2004.
4. A K Sharma , "Object Oriented Programming with C++", Pearson Education, 2014


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

1. Basics of C++ - https://www.youtube.com/watch?v=BClS40yzssA
2. Functions of C++ - https://www.youtube.com/watch?v=p8ehAjZWjPw

**Tutorial Link:**

1. https://www.w3schools.com/cpp/cpp_intro.asp
2. https://www.edx.org/course/introduction-to-c-3
3. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013843642506788864433 75_shared/overview

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

● Group Assignment to develop small projects and demonstrate using C++

| DATA ANALYTICS WITH EXCEL | | | |
|---|---|---|---|
| **Course Code** | 25ISL37A | **CIE Marks** | 50 |
| **Teaching Hours/Week (L:T:P)** | 0:0:2 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 28 | **Exam Hours** | 2 |
| **Examination type (SEE)** | Practical | | |

**Prerequisites:**

The students should have knowledge of

- Basic Excel Skills
- Knowledge of Basic Statistics and Logic
- Basic Knowledge of Pivot Tables
- Familiarity with Charts and Visualizations

**Course Objectives:**

**CLO1:**To Apply analysis techniques to datasets in Excel
**CLO2:**Learn how to use Pivot Tables and Pivot Charts to streamline your workflow in Excel
**CLO3:**Understand and Identify the principles of data analysis
**CLO4:**Become adept at using Excel functions and techniques for analysis
**CLO5:**Build presentation ready dashboards in Excel.

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**LIST OF PROGRAMS**

1. Getting Started with Excel: Creation of spread sheets, Insertion of rows and columns, Drag & Fill, use of Aggregate functions.
2. Working with Data : Importing data, Data Entry & Manipulation, Sorting & Filtering.
3. Working with Data: Data Validation, Pivot Tables & Pivot Charts.
4. Data Analysis Process: Conditional Formatting, What-If Analysis, Data Tables, Charts & Graphs.
5. Cleaning Data with Text Functions: use of UPPER and LOWER, TRIM function, Concatenate.
6. Cleaning Data Containing Date and Time Values: use of DATEVALUE function, DATEADD and DATEDIF, TIMEVALUE functions.
7. Conditional Formatting: formatting, parsing, and highlighting data in spreadsheets during data analysis.
8. Working with Multiple Sheets: work with multiple sheets within a workbook is crucial for organizing

and managing data, perform complex calculations and create comprehensive reports.

9. Create worksheet with following fields: Empno, Ename, Basic Pay(BP), Travelling Allowance(TA), Dearness Allowance(DA), House Rent Allowance(HRA), Income Tax(IT), Provident Fund(PF), Net Pay(NP). Use appropriate formulas to calculate the above scenario. Analyse the data using appropriate chart and report the data.
10. Create worksheet on Inventory Management: Sheet should contain Product code, Product name, Product type, MRP, Cost after % of discount, Date of purchase. Use appropriate formulas to calculate the above scenario. Analyse the data using appropriate chart and report the data.
11. Create worksheet on Sales analysis of Merchandise Store: data consisting of Order ID, Customer ID, Gender, age, date of order, month, online platform, Category of product, size, quantity, amount, shipping city and other details. Use of formula to segregate different categories and perform a comparative study using pivot tables and different sort of charts.
12. Generation of report & presentation using Autofilter &macro.

**Course outcomes (Course Skill Set):**

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

**CO1:** Use advanced functions and productivity tools to assist in developing worksheets.(PO-1,3,5, PSO-1,3)
**CO2:** Manipulate data lists using Outline and PivotTables. (PO-1,4,5, PSO-1,2)
**CO3:** Use Consolidation to summarize and report results from multiple worksheets. (PO-1,4,5, PSO-1,2)
**CO4:** Apply Macros and Auto filter to solve the given real-world scenario. ( PO-1,3,5, PSO-1,3)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- CIE marks for the practical course are 50 Marks. The split-up of CIE marks for record/ journal and test are in the ratio 60:40.
- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to 20 marks (40% of the maximum marks). The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student

• Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**Semester-End Examination:**

SEE marks for the practical course are 50 Marks.

● SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.

● The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University

All laboratory experiments are to be included for practical examination.

● (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.

● Students can pick one question from PART-A and one question from PART-B (experiment) from the questions lot prepared by the examiners jointly and 50% weightage for each part.

● Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

● General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners) Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero. The minimum duration of SEE is 02 hours

**Suggested Learning Resources:**

**Text Books:**

1. S. Sridhar, J. Indumathi, V.M. Hariharan "Python Programming" Pearson publishers, 1st edition 2023.
2. Fabio Nelli, "Python Data Analytics", Apress, Publishing, 1st Edition, 2015.

**Reference Book:**

1. Paul Deitel and Harvey deitel,"Intro to Python for Computer Science and Data science", 1st edition Pearson Publisher 2020.

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

- https://ds1-iiith.vlabs.ac.in/exp/stacks-queues/index.html
- https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html
- https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html
- https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/index.html
- https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/depth-first-traversal/dft-practice.html

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

- Assessment Methods
  - Programming Assignment (10 Marks)

| Project Management with Git | | | |
|---|---|---|---|
| **Course Code** | 25ISL37B | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 0:0:2 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 28 | **Exam Hours** | 2 |
| **Examination nature (SEE)** | Practical | | |

**Prerequisites:** NIL

**Course Objectives:**

**CLO1:** To become familiar with basic Git commands for version control.
**CLO2:** To create and manage branches for effective code organization.
**CLO3:** To understand how to collaborate and work with remote repositories.
**CLO4:** To master version control commands for efficient project management.

**List of Experiments:**

| S. No | Experiments |
|---|---|
| 1 | **Setting up and using basic Git commands**<br>• Create a new repository and configure it with essential settings.<br>• Track changes to files by adding them to the staging area and committing with descriptive messages. |
| 2 | **Creating and Managing Branches**<br>Create a new branch named "feature-branch." Switch to the "master" branch. Merge the "feature-branch" into "master." |
| 3 | **Merging and Conflict Resolution**<br>Write the commands to merge a branch into the main branch and resolve any merge conflicts. |
| 4 | **Collaboration and Remote Repositories**<br>Clone a remote Git repository to your local machine. |
| 5 | **Collaboration and Remote Repositories**<br>• Fetch the latest changes from a remote repository and rebase your local branch onto the updated remote branch.<br>• Write the command to merge "feature-branch" into "master" while providing a custom commit message for the merge. |
| 6 | **Pulling Changes**<br>Write the command to pull the latest changes from the remote "origin" repository. |
| 7 | **Git Tags and Releases** |

| | |
|---|---|
| | Write the command to create a lightweight Git tag named "v1.0" for a commit in your local repository. |
| 8 | **Advanced Git Operations**<br>Write the command to cherry-pick a range of commits from "source-branch" to the current branch. |
| 9 | **Analysing and Changing Git History**<br>Given a commit ID, how would you use Git to view the details of that specific commit, including the author, date, and commit message? |
| 10 | **Reviewing and Modifying Git Commit History**<br>Write the command to list all commits made by the author "Pressman" between "2024-01-01" and "2024-12-31." |
| 11 | **Inspecting and Editing Git History**<br>Write the command to display the last five commits in the repository's history. |
| 12 | **Exploring and Refining Git Commit Logs**<br>Write the command to undo the changes introduced by the commit with the ID "xyz12345". |

**Course outcomes (Course Skill Set):**

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

**CO1:** Use the basic commands related to git repository. (PO – 1, 2, 3, 5, 8, 11, PSO – 1, 2, 3)

**CO2:** Create and manage the branches. (PO – 1, 2, 3, 5, 8, 11, PSO – 1, 2, 3)

**CO3:** Apply commands related to collaboration and remote repositories. (PO – 1, 2, 3, 5, 8, 11, PSO – 1, 2, 3)

**CO4:** Use the commands related to Git tags, releases and advanced git operations. (PO – 1, 2, 3, 5, 8, 11, PSO – 1, 2, 3)

**CO5:** Analyse and change the git history. (PO – 1, 2, 3, 5, 8, 11, PSO – 1, 2, 3)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation (CIE):**

CIE marks for the practical course are 50 Marks.

The split-up of CIE marks for record/ journal and test are in the ratio 60:40.

- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be

evaluated for 10 marks.

- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to 20 marks (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student.

**Semester End Evaluation (SEE):**

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted by the two examiners. One from the same institute as an internal examiner and another from a different institute as an external examiner, appointed by the university.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University.
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in - 60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)
- Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero.

The minimum duration of SEE is 02 hours.

**Suggested Learning Resources:**


- "Version Control with Git", 3rd Edition, by Prem Kumar Ponuthorai, Jon Loeliger Released October 2022, Publisher(s): O'Reilly Media, Inc.
- "Pro Git book", written by Scott Chacon and Ben Straub and published by Apress, https://gitscm.com/book/en/v2
- https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013094443347369984278 2_shared/overview
- https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013301347121774592119 26_shared/overview

| PROMPT ENGINEERING | | | |
|---|---|---|---|
| **Course Code** | 25ISL37C | **CIE Marks** | 50 |
| **Teaching Hours/Week (L:T:P)** | 0:0:2 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 28 | **Exam Hours** | 2 |
| **Examination type (SEE)** | Practical | | |

**Prerequisite:**

**The students should have knowledge on**

- Knowledge on Basic Programming Skills (Python preferred)
- Knowledge on Comfort with Using Chatbots or AI Tools
- Knowledge on Internet Research Skills
- Knowledge on Understanding of Ethics in AI Usage

**Course objectives:**

**CLO1:**Understand the fundamentals and types of prompts used in generative AI models.
**CLO2:**Explore the role of prompt engineering in shaping AI outputs and model behaviour.
**CLO3:**Gain hands-on skills in designing, optimizing, and validating prompts for NLP and AI tasks.
**CLO4:**Familiarize themselves with advanced prompting techniques such as Chain-of-Thought and React.
**CLO5:**Apply prompt engineering responsibly by considering ethical implications and quality benchmarks.

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

| Sl.NO | Experiment Description |
|---|---|
| 1 | **Design and compare different types of prompts** including zero-shot, one-shot, few-shot, and instructional prompts using a generative AI model. |
| 2 | **Evaluate and validate prompt quality** by testing multiple prompt variants for the same task and analysing output consistency and clarity. |
| 3 | **Develop prompts for common NLP tasks** such as sentiment analysis, summarization, translation, and text classification. |
| 4 | **Optimize a basic prompt** by refining it step-by-step using prompt tuning techniques to improve response accuracy and relevance. |

| | |
|---|---|
| 5 | **Test the same prompt across multiple generative AI models** (e.g., ChatGPT, Claude, Gemini) and analyse variations in output. |
| 6 | **Use the OpenAI Chat Completion API** to simulate a structured interaction using system, user, and assistant roles. |
| 7 | **Compare outputs from Instruct and ChatGPT** for the same prompts and identify Behavioral differences in response generation. |
| 8 | **Apply the CLEAR Framework** (Concise, Logical, Explicit, Adaptive, Reflective) to improve poorly performing prompts. |
| 9 | **Analyse the impact of small prompt changes** on output by modifying wording, tone, or context and documenting the effect. |
| 10 | **Implement Chain-of-Thought (CoT) prompting** for solving multi-step reasoning or problem-solving tasks. |
| 11 | **Use Tree-of-Thoughts or React prompting techniques** to model step-by-step decision-making in complex tasks. |
| 12 | **Explore Automatic Prompt Engineering (APE)** by using tools or code to automatically generate or refine effective prompts. |

**Course outcomes (Course Skill Set):**

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

**CO1:** Identify and classify different types of prompts and their use cases in AI systems. (PO -1,2,3,5, PSO -1,2,3)
**CO2:** Develop and evaluate prompts for various NLP and generative AI applications. (PO -1,2,3,4,5, PSO -1,2,3)
**CO3:** Use APIs and tools to implement prompt-based solutions and analyse output consistency. PO -1,2,3,4,5, PSO - 1,2,3)
**CO4:** Apply frameworks and strategies to craft effective and adaptive prompts. (PO -1,2,3,5, PSO -1,2,3)
**CO5:** Demonstrate the ability to use advanced prompt techniques and ethical practices in AI. (PO -1,2,3,5,7, PSO - 1,2,3)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together

**Continuous Internal Evaluation (CIE):**

CIE marks for the practical course are 50 Marks.

The split-up of CIE marks for record/ journal and test are in the ratio 60:40.

- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be

evaluated for 10 marks.

● Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).

● Weightage to be given for neatness and submission of record/write-up on time.

● Department shall conduct 02 tests for 100 marks, the first test shall be conducted after the 8th week of the semester and the second test shall be conducted after the 14th week of the semester.

● In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.

● The suitable rubrics can be designed to evaluate each student's performance and learning ability.

● The average of 02 tests is scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student

**Semester End Evaluation (SEE):**

● SEE marks for the practical course are 50 Marks.

● All laboratory experiments are to be included for practical examination.

● (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.

● Students can pick one question (experiment) from the questions lot prepared by the examiners jointly.

● Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

● General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in 60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks. Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero.

**Suggested Learning Resources:**

| | | |
|---|---|---|
| **ChatGPT** (OpenAI) | Generative AI model for prompt testing | https://chat.openai.com |
| **OpenAI API Platform** | Programmatic access to GPT models | https://platform.openai.com |
| **Claude (Anthropic)** | LLM for safe and interpretable prompts | https://claude.ai |
| **Google Gemini (Bard)** | AI chatbot by Google | https://gemini.google.com |
| **Google Colab** | Cloud-based Jupyter notebook | https://colab.research.google.com |
| **Prompt Perfect** | Prompt optimization tool | https://promptperfect.jina.ai |
| **Flow GPT** | Prompt sharing and community exploration | https://flowgpt.com |
| **Prompt Base** | Marketplace of engineered prompts | https://promptbase.com |
| **Lang Chain** | Framework for LLM-based applications | https://www.langchain.com |
| **Llama Index** | Data framework for connecting LLMs | https://www.llamaindex.ai |
| **Prompt Layer** | Logs, tracks, and evaluates prompt usage via API | https://www.promptlayer.com |

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

- Design and test effective AI prompts through hands-on tasks.
- Analyse and refine outputs via iterative improvement.
- Engage in collaborative AI problem-solving activities.

IV Semester

**BE in Information Science & Engineering**

**Scheme of Teaching IV Semester**

Outcome Based Education and Choice Based Credit System (CBCS)

Effective from the academic year Batch - 2025-29

| Sl. No | Course | Course Code | Course Title | Teaching Department (TD)/ Board | Teaching Hours /Week | | | TW & SL | TH/S | Examination | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | L | T | P | | | CIE Marks | SEE Marks | Total Marks | Credits |
| 1 | PCC | 25IS41 | Analysis and Design of Algorithms | ISE | 3 | 0 | 0 | 48 | 90 | 50 | 50 | 100 | 3 |
| 2 | PCC | 25IS42 | Data Base Management System | ISE | 3 | 0 | 0 | 48 | 90 | 50 | 50 | 100 | 3 |
| 3 | PCC | 25IS43 | Software Engineering | ISE | 3 | 0 | 0 | 48 | 90 | 50 | 50 | 100 | 3 |
| 4 | IPCC | 25IS44 | Fundamentals of Cloud Computing | ISE | 3 | 0 | 2 | 50 | 120 | 50 | 50 | 100 | 4 |
| 5 | PCCL | 25ISL45 | Analysis and Design of Algorithms Lab | ISE | 0 | 0 | 2 | 02 | 30 | 50 | 50 | 100 | 1 |
| 6 | BSC/ ESC | 25IS46X | **BSC/ESC** | Mathematics | 2 | 2 | 0 | 34 | 90 | 50 | 50 | 100 | 3 |
| 7 | AEC | 25ISL47X | Ability Enhancement Course-IV | ISE dept/ IT Industry | 0 | 0 | 2 | 02 | 30 | 50 | 50 | 100 | 1 |
| 8 | UHV | 25UH48 | Universal Human Values | Any dept | 1 | 0 | 0 | 16 | 30 | 50 | 50 | 100 | 1 |
| 9 | NCMC | 25NS49/ 25PE49/ 25YO49 | NSS/YOGA/ PE | NSS/ YOGA/ PE Coordinator | 0 | 0 | 0 | 28 | 28 | 100 | - | 100 | PP |
| 10 | SDC | 25SC410 | Environmental Science Project | | 0 | 0 | 0 | 30 | 30 | 50 | 50 | 100 | 1 |
| **Total** | | | | | | | | | **628** | **550** | **450** | **1000** | **20** |
| 11 | NCMC | 25MDIP411 | Mathematics course for Lateral Entry Students | Maths Dept | 1 | 0 | 0 | 14 | 28 | 100 | - | 100 | PP |
| 12 | SK | 25ISSK412 | Skill Lab-2: Mobile Application Development | ISE | 0 | 0 | 2 | 02 | 30 | 50 | 50 | 100 | 1 |
| **Total** | | | | | | | | | **686** | **700** | **500** | **1200** | **20+1** |

**Note:**

L: Lecture , T: Tutorial, P: Practical

TW & SL: Term Work & Self Learning

TH/S: Total Hous/Semester

| **BSC/Engineering Science Course (ESC/ETC/PLC)– IV** | | | |
|---|---|---|---|
| 25IS46A | Linear Algebra | 25IS46C | Graph Theory |
| 25IS46B | Discrete Mathematics structures | | |

| **Ability Enhancement Course / Skill Enhancement Course - IV** | | | |
|---|---|---|---|
| 25ISL47A | UI/UX Design Tool | 25ISL47C | Technical writing using LATEX |
| 25ISL47B | DBMS with SQL and MongoDB | | |

| ANALYSIS AND DESIGN OF ALGORITHMS | | | |
|---|---|---|---|
| **Course Code** | 25IS41 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 3:0:0 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 42 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge on

- Fundamentals of Data Structures
- Understanding of Algorithms
- Basic Mathematics
- Problem-Solving Skills

**Course Objectives:**

CLO1: To learn the methods for analyzing algorithms and evaluating their performance.
CLO2: To demonstrate the efficiency of algorithms using asymptotic notations.
CLO3: To solve problems using various algorithm design methods, including brute force, greedy, divide and conquer, decrease and conquer, transform and conquer, dynamic programming, backtracking, and branch and bound.
CLO4: To learn the concepts of P and NP complexity classes.

**Teaching-Learning Process (General Instructions)**

**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) does not mean only the traditional lecture method, but different types of teaching methods may be adopted to achieve the outcomes.
2. Utilize video/animation films to illustrate the functioning of various concepts.
3. Promote collaborative learning (Group Learning) in the class.
4. Pose at least three HOT (Higher Order Thinking) questions in the class to stimulate critical thinking.
5. Incorporate Problem-Based Learning (PBL) to foster students' analytical skills and develop their ability to evaluate, generalize, and analyze information rather than merely recalling it.
6. Introduce topics through multiple representations.
7. Demonstrate various ways to solve the same problem and encourage students to devise their own creative solutions.
8. Discuss the real-world applications of every concept to enhance students' comprehension

| **Module-1** | **9 Hours** |
|---|---|

**INTRODUCTION:** What is an Algorithm? Fundamentals of Algorithmic Problem Solving.
**FUNDAMENTALS OF THE ANALYSIS OF ALGORITHM EFFICIENCY:** Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non recursive Algorithms, Mathematical Analysis of Recursive Algorithms.
**BRUTE FORCE APPROACHES:** Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching.
**Chapter 1 (Sections 1.1,1.2), Chapter 2(Sections 2.1,2.2,2.3,2.4), Chapter 3(Section 3.1,3.2)**

| **Module-2** | **9 Hours** |
|---|---|

**BRUTE FORCE APPROACHES (contd..):** Exhaustive Search (Travelling Salesman problem and Knapsack Problem).
**DECREASE-AND-CONQUER:** Insertion Sort, Topological Sorting.
**DIVIDE AND CONQUER:** Merge Sort, Quick Sort, Binary Tree Traversals, Multiplication of Large Integers and Strassen's Matrix Multiplication.

**Module-3** **8 Hours**

**TRANSFORM-AND-CONQUER:** Balanced Search Trees, Heaps and Heapsort.
**SPACE-TIME TRADEOFFS:** Sorting by Counting: Comparison counting sort, Input Enhancement in String Matching: Horspool's Algorithm.
**Chapter 6 (Sections 6.3,6.4), Chapter 7 (Sections 7.1,7.2)**

**Module-4** **8 Hours**

**DYNAMIC PROGRAMMING:** Three basic examples, The Knapsack Problem and Memory Functions, Warshall's and Floyd's Algorithms.
**THE GREEDY METHOD:** Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees and Codes.
**Chapter 8 (Sections 8.1,8.2,8.4), Chapter 9 (Sections 9.1,9.2,9.3,9.4)**

**Module-5** **8 Hours**

**LIMITATIONS OF ALGORITHMIC POWER:** Decision Trees, P, NP, and NP-Complete Problems.
**COPING WITH LIMITATIONS OF ALGORITHMIC POWER**: Backtracking (n-Queens problem, Subset-sum problem), Branch-and-Bound (Knapsack problem), Approximation algorithms for NP-Hard problems (Knapsack problem).
**Chapter 11 (Section 11.2, 11.3), Chapter 12 (Sections 12.1,12.2,12.3)**

**Course outcome (Course Skill Set)**

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

1. Apply asymptotic notational method to analyze the performance of the algorithms in terms of time complexity. (PO 1,2,3,4)
2. Demonstrate divide & conquer approaches and decrease & conquer approaches to solve computational problems. (PO 1,2,3,4)
3. Make use of transform & conquer and dynamic programming design approaches to solve the given real world or complex computational problems. (PO 1,2,3,4,5)
4. Apply greedy and input enhancement methods to solve graph & string based computational problems. (PO 1,2,3,4,5)
5. Analyse various classes (P,NP and NP Complete) of problems Illustrate backtracking, branch & bound and approximation methods. (PO 1,2,3,4,5)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:**

Theory SEE will be conducted by the University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.

Marks scored shall be proportionally **reduced to 50 marks**

**Suggested Learning Resources:**

**Textbooks**

1. Introduction to the Design and Analysis of Algorithms, By Anany Levitin, 3rd Edition (Indian), 2017, Pearson.

**Reference books**

1. Computer Algorithms/C++, Ellis Horowitz, Sartaj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press.
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.

Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education)

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

- https://www.youtube.com/watch?v=tSo9yzBUTHA
- https://www.youtube.com/watch?v=5_5oE5lgrhw
- https://www.youtube.com/watch?v=STL8ESuETmM
- https://www.youtube.com/watch?v=Y2A8RzxegSA
- https://www.youtube.com/watch?v=gBz44smaa9A
- https://www.youtube.com/watch?v=8TVaEGeaGGc
- https://www.youtube.com/watch?v=kPRA0W1kECg
- https://www.youtube.com/watch?v=v0eQ4nXJjsk
- https://www.youtube.com/watch?v=MtQL_ll5KhQ
- https://www.youtube.com/watch?v=jsmMtJpPnhU
- https://www.youtube.com/watch?v=tKwnms5iRBU
- https://www.youtube.com/watch?v=oBt53YbR9Kk
- https://www.youtube.com/watch?v=OQ5jsbhAv_M
- https://www.youtube.com/watch?v=Aa2sqUhIn-E

- https://www.youtube.com/watch?v=SZXXnB7vSm4
- https://www.youtube.com/watch?v=MHXR4PCY8c0
- https://www.youtube.com/watch?v=kJuKy5FqhrE
- https://www.youtube.com/watch?v=TtuNf6XMhiw
- Design and Analysis of Algorithms: https://nptel.ac.in/courses/106/101/106101060/

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

- Promote real-world problem-solving and competitive problem solving through group discussions to engage students actively in the learning process.

- Encourage students to enhance their problem-solving skills by implementing algorithms and solutions through programming exercises, fostering practical application of theoretical concepts.

**Assessment Methods -**

1. Problem Solving Assignments (Hacker Rank/ Hacker Earth / Leadcode)
    - Gate Based Aptitude Test

| DATABASE MANAGEMENT SYSTEMS | | | |
|---|---|---|---|
| **Course Code** | 25IS42 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 3:0:0 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 42 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

Students should have a basic understanding of programming concepts, data structures, computer organization, and operating systems. Familiarity with software development principles and problem-solving techniques is desirable for studying database design, SQL, transaction management, and MongoDB-based application development.

**Course Objectives:**

**CLO1:** Understand the fundamental concepts of database systems, data models, database architecture, schemas, and Entity-Relationship (ER) modelling.

**CLO2:** Learn the principles of the relational data model, relational algebra, and database schema design techniques.

**CLO3:** Apply various Structured Query Language (SQL) statements for database manipulation.

**CLO4:** Understand normalization techniques and transaction management to design efficient, reliable, and consistent database systems.

**CLO5:** To provide knowledge of MongoDB concepts and database operations.

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt **Problem-Based Learning (PBL)** by assigning real-world database design and querying problems.
6. Demonstrate database application development using **MongoDB and the Node.js Driver** through practical examples and mini-projects.

**Module-1** **9 Hours**

Introduction to Databases: Introduction, Characteristics of database approach, Advantages of using the DBMS approach, History of database applications. Overview of Database Languages and Architectures: Data Models, Schemas, and Instances. Three schema architecture and data independence, 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.

**Textbook 1**:Ch 1.1 to 1.8, 2.1 to 2.6, 3.1 to 3.10

**Module-2** **8 Hours**

Relational Model: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations. Relational Algebra: Unary and Binary relational operations, additional relational operations (aggregate, grouping, etc.) Examples of Queries in relational algebra. Mapping Conceptual Design into a Logical Design: Relational Database

Design using ER-to-Relational mapping.

**Textbook 1**: Ch 5.1 to 5.3, Ch 8.1 to 8.5; Ch 9.1 to 9.2 Textbook 2: 3.5

**Module-3** **8 Hours**

SQL: SQL data definition and data types, Schema change statements in SQL,SQL Languages (DML,DDL,TCL) specifying constraints in SQL, retrieval queries in SQL, INSERT, DELETE, and UPDATE statements in SQL, Additional features of SQL.

SQL: Advanced Queries: More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Cursor , Stored Procedure, Views in SQL.

**Textbook 1:** Ch 6.1 to 6.5 and Ch 7.1 to 7.3, Ch 10.2.2 and 10.4.1

**Module-4** **9 Hours**

Normalization: Database Design Theory — Introduction to Normalization using Functional and Multivalued Dependencies: Informal design guidelines for relation schema, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

Transaction Processing: Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Concurrency Control in Databases: Two-phase locking techniques for Concurrency control.

**Textbook 1:** Ch 14.1 to 14.7 and Ch 20.1 to 20.6

**Module-5** **8 Hours**

MongoDB: Basics, Documents, Collections, Databases, Query Language, Installation, The Mongo Shell, MongoDB CRUD Operations, Create, Read, Projection, Update, Delete, Aggregate, MongoDB Node.js Driver, Schema Initialization.

**Textbook 2:** Chapter 6

**Course outcome (Course Skill Set):**

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

**CO1:** Describe database concepts, data models, schemas, and ER modelling. (PO – 1,2,3,4,5, PSO – 1,2,3)
**CO2:** Apply relational model concepts and relational algebra for schema design. (PO – 1,2,3,4,5, PSO – 1,2,3)
**CO3:** Use SQL for database creation, manipulation, and advanced querying. (PO – 1,2,3,4,5, PSO – 1,2,3)
**CO4:** Normalize relations and explain transaction management techniques. (PO – 1,2,3,4,5, PSO – 1,2,3)
**CO5:** Implement MongoDB databases using CRUD operations, aggregation, and application integration. (PO – 1,2,3,4,5, PSO – 1,2,3)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation(CIE):**

● For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment

Test component, there are 25 marks.

● The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

● Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.

● For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**Semester-End Examination:**

Theory SEE will be conducted by the University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**).

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks.

**Suggested Learning Resources:**

**TEXTBOOKS:**


1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. Vasan Subramanian, Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node. Apress, 2019.


**REFERENCES:**


1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, *Database System Concepts*, McGraw Hill, 7th Edition, 2020.
2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill.
3. C.J. Date, *An Introduction to Database Systems*, Pearson Education, 8th Edition.
4. Amol Nayak, *MongoDB for Developers*, Packt Publishing.


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

- https://www.youtube.com/watch?v=GgznCfPjXOk
- https://www.youtube.com/watch?v=I1zbwBqgOQY
- https://www.youtube.com/watch?v=ryeGFOMZhK4
- https://www.youtube.com/watch?v=WK8MHedLFmk
- https://www.youtube.com/watch?v=IBEuwmYsMKI
- https://www.youtube.com/watch?v=rBmo2UAa8Lk
- https://www.youtube.com/watch?v=EjcAqAJjmEo

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

Assessment Methods for 25 Marks (opt two Learning Activities)

- Case Study
- Mini Project
- Quiz

| SOFTWARE ENGINEERING | | | |
|---|---|---|---|
| **Course Code** | 25IS43 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L:T:P)** | 3:0:0 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 42 | **Exam Hours** | 3 |
| **Examination nature (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge on

- Object-Oriented Programming (OOP) concepts
- Data Structures and Algorithms fundamentals
- Basics of Software Development Tools

**Course objectives:**

**CLO1:**To introduce fundamental concepts of software engineering, software process models, and requirements engineering for systematic software development.

**CLO2:**To develop an understanding of object-oriented concepts and modelling techniques for representing software systems.

**CLO3:**To familiarize students with different system models, object-oriented design principles, UML, design patterns, and implementation approaches.

**CLO4:**To develop knowledge of software testing techniques and software evolution activities for ensuring software quality and maintainability.

**CLO5:**To introduce software project planning, estimation, quality management, measurement, metrics, and software standards.

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**MODULE-1** **09 Hours**

Introduction: Software Crisis, Need for Software Engineering. Professional Software Development, Software Engineering Ethics. Case Studies. Software Processes: Models: Waterfall Model (Sec 2.1.1), Incremental Model (Sec 2.1.2) and Spiral Model (Sec 2.1.3). Process activities. Requirements Engineering: Requirements Engineering Processes (Chap 4). Requirements Elicitation and Analysis (Sec 4.5). Functional and non-functional requirements (Sec 4.1). The software Requirements Document (Sec 4.2). Requirements Specification (Sec 4.3). Requirements validation (Sec 4.6). Requirements Management (Sec 4.7).

**Textbook 1**: Chapter 2-2.1.1 to 2.1.3, Chapter 4 4.1 to 4.3 , 4.4 to 4.7

**MODULE-2** **09 Hours**

What is Object orientation? What is OO development? OO Themes; Evidence for usefulness of OO development; OO modelling history. Modelling as Design technique: Modelling; abstraction; The Three models. Introduction, Modelling Concepts and Class Modelling: What is Object orientation? What is OO development? OO Themes; Evidence for usefulness of OO development; OO modelling history. Modelling as Design technique: Modelling; abstraction; The Three models. Class Modelling: Object and Class Concept; Link and associations concepts; Generalization and Inheritance; A sample class model; Navigation of class models;
**Textbook 2**: Chapter 1,2,3.

**MODULE-3** **08 Hours**

**System Models**: Context models (Sec 5.1). Interaction models (Sec 5.2). Structural models (Sec 5.3). Behavioral models (Sec 5.4). Model-driven engineering (Sec 5.5). Design and Implementation: Introduction to RUP (Sec 2.4), Design Principles (Chap 7). Object-oriented design using the UML (Sec 7.1). Design patterns (Sec 7.2). Implementation issues (Sec 7.3). Open source development (Sec 7.4).
**Textbook 1:** Chapter 5.1 to 5.5, 7.1 to 7.4

**MODULE-4** **08 Hours**

**Software Testing**: Development testing (Sec 8.1), Test-driven development (Sec 8.2), Release testing (Sec 8.3), User testing (Sec 8.4). Test Automation (Page no 212). Software Evolution: Evolution processes (Sec 9.1). Program evolution dynamics (Sec 9.2). Software maintenance (Sec 9.3). Legacy system management (Sec 9.4).
**Textbook 1:** Chapter 8.1 to 8.4, 9.1 to 9.4

**MODULE-5** **08 Hours**

**Project Planning**: Software pricing (Sec 23.1). Plan-driven development (Sec 23.2). Project scheduling (Sec 23.3): Estimation techniques (Sec 23.5). Quality management: Software quality (Sec 24.1). Reviews and inspections (Sec 24.3). Software measurement and metrics (Sec 24.4). Software standards (Sec 24.2)
**Textbook 1:** Chapter 23.1 to 23.3,23.5, 24.1 to 24.4

**Course outcome (Course Skill Set**)

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

**CO1:** Explain software engineering principles, software process models, and requirements engineering activities.
**CO2:** Apply object-oriented concepts and modelling techniques to develop class models for software systems.
**CO3:** Design software systems using appropriate system models, UML-based object-oriented design principles, design patterns, and implementation approaches.
**CO4:** Apply software testing techniques and explain software evolution and maintenance processes.
**CO5:** Analyze software project planning and quality management practices using appropriate estimation, metrics, and standards.

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation(CIE):**

● For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.

● The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

● Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two

assignments at the end of the semester if two assignments are planned.

● For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:** Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

**Suggested Learning Resources:**

**Textbooks**

1. Ian Sommerville: Software Engineering, 9th Edition, Pearson Education, 2012. (Listed topics only from Chapters 1,2,3,4, 5, 7, 8, 9, 23, and 24)
2. Michael Blaha, James Rumbaugh: Object Oriented Modelling and Design with UML,2nd Edition, Pearson Education,2005.

**Reference Books:**

1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill.
2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India

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

- https://www.youtube.com/playlist?list=PLPzfPcir5uPR8vKjOFitZZdEIT8-snomq
- https://www.youtube.com/watch?v=6XrL5jXmTwM
- https://www.youtube.com/watch?v=yfP_v6qCdcs
- https://www.youtube.com/watch?v=VjNH0MuC2iU
- https://www.youtube.com/watch?v=rtYoMUi5vU4

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

- Role Play
- Flipped classroom

| FUNDAMENTALS OF CLOUD COMPUTING | | | |
|---|---|---|---|
| **Course Code** | 25IS44 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T:P)** | 3:0:2 | **SEE Marks** | 50 |
| **Credits** | 04 | **Total Marks** | 100 |
| **Contact Hours** | 42 Theory + 28 Practical | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge on

- Basics of Computer Networks
- Operating System Fundamentals
- Understanding of Web Technologies

**Course objectives:**

- **CLO1:** Introduce the rationale behind the cloud computing revolution and the business drivers
- **CLO2:** Understand various models, types and challenges of cloud computing
- **CLO3:** Understand the design of cloud native applications, the necessary tools and the design tradeoffs.
- **CLO4:** Realize the importance of Cloud Virtualization, Abstraction`s, Enabling Technologies and cloud security

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Module-1** **08 Hours**

**Distributed System Models and Enabling Technologies:** Scalable Computing Over the Internet, Technologies for Network Based Systems, System Models for Distributed and Cloud Computing, Software Environments for Distributed Systems and Clouds, Performance, Security and Energy Efficiency.

**Textbook 1: Chapter 1: 1.1 to 1.5**

**Module-2** **08 Hours**

**Virtual Machines and Virtualization of Clusters and Data Centers:** Implementation Levels of Virtualization, Virtualization Structure/Tools and Mechanisms, Virtualization of CPU/Memory and I/O devices, Virtual Clusters and Resource Management, Virtualization for Data Center Automation.

**Textbook 1: Chapter 3: 3.1 to 3.5**

**Module 3** **09 Hours**

**Cloud Platform Architecture over Virtualized Datacenters:** Cloud Computing and Service Models, Data Center Design and Interconnection Networks, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms: GAE, AWS and Azure, Inter-Cloud Resource Management.

**Textbook 1: Chapter 4: 4.1 to 4.5**

| Module-4 | 08 Hours |
|---|---|

**Cloud Security:** Top concern for cloud users, Risks, Privacy Impact Assessment, Cloud Data Encryption, Security of Database Services, OS security, VM Security, Security Risks Posed by Shared Images and Management OS, XOAR, A Trusted Hypervisor, Mobile Devices and Cloud Security

**Cloud Security and Trust Management:** Cloud Security Defense Strategies, Distributed Intrusion/Anomaly Detection, Data and Software Protection Techniques, Reputation-Guided Protection of Data Centers.

**Textbook 2: Chapter 11: 11.1 to 11.3, 11.5 to 11.8, 11.10 to 11.14**
**Textbook 1: Chapter 4: 4.6**

| Module-5 | 09 Hours |
|---|---|

**Cloud Programming and Software Environments:**
Features of Cloud and Grid Platforms, Parallel and Distributed Computing Paradigms, Programming Support for Google App Engine, Programming on Amazon AWS and Microsoft, Emerging Cloud Software Environments.
**Textbook 1: Chapter 6: 6.1 to 6.5**

**PRACTICAL COMPONENT**

1.Creating a Virtual Machine: Configure and deploy a virtual machine with specific CPU and memory requirements in Google Cloud. OR Exploring AWS CloudShell and the AWS Cloud9 IDE

2. Getting Started with Cloud Shell and gcloud: Discover the use of gcloud commands to manage Google Cloud resources from Cloud Shell. OR Working with Amazon S3Orchestrating Serverless Functions with AWS Step Functions

3. Cloud Functions: Create and deploy a Cloud Function to automate a specific task based on a Cloud Storage event OR Working with Amazon DynamoDB

4. App Engine: Deploy a web application on App Engine with automatic scaling enabled. OR Developing REST APIs with Amazon API Gateway

5. Cloud Storage: Qwikstart: Google Cloud Storage provides scalable and secure object storage for managing data, accessible via the Cloud Console or gsutil CLI.
OR Creating Lambda Functions Using the AWS SDK for Python

6. Cloud SQL for MySQL: Discover how Google Cloud SQL for MySQL provide automated management and high availability for MySQL databases?
OR Migrating a Web Application to Docker Containers

7. Cloud Pub/Sub: Experiment how Google Cloud Pub/Sub facilitate real-time messaging and communication between distributed applications.
OR Caching Application Data with ElastiCache, Caching with Amazon CloudFronT, Caching Strategies

8. Multiple VPC Networks: Explore benefits of using multiple VPC networks in Google Cloud for organizing and isolating resources.
OR Implementing CloudFront for Caching and Application Security

9.Cloud Monitoring: Discover how Cloud Monitoring help in tracking and analyzing the performance and health

of cloud resources?
OR
Orchestrating Serverless Functions with AWS Step Functions

10. Kubernetes Engine: Qwik Start: Deploy a containerized application to a Kubernetes Engine cluster.
OR Automating Application Deployment Using a CI/CD Pipeline

11. Complex Experiments (Not for CIE)

**1.** Create and Manage Cloud Resources: Challenge Lab: In this lab, Students will use the Google Cloud Console and the gcloud command-line tool to create and manage various cloud resources. Start by provisioning virtual machines with specific configurations, such as CPU and memory requirements, and setting up storage buckets for data persistence. Students also manage IAM roles to control access to these resources, ensuring that only authorized users can perform actions. The lab emphasizes the importance of understanding the relationships between different Google Cloud services and how to configure them to work together effectively. Successful completion requires a careful approach to resource management, including monitoring, security settings, and cost optimization.

**2.** Set Up an App Dev Environment on Google Cloud: Challenge Lab:

This lab focuses on setting up a complete development environment on Google Cloud, starting with configuring Cloud Shell and installing the necessary development tools. Students work with Cloud SDK and other programming languages or frameworks required for your application. After setting up the environment, Deploy a sample application to test the configuration and ensure that the environment is fully functional. This lab highlights the importance of creating a robust and scalable environment that can support continuous development and deployment processes. Additionally, you must ensure that the environment is optimized for performance and ready to handle real-world application development and testing on Google Cloud.

**Course outcome (Course Skill Set)**

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

**CO1:** Describe various cloud computing platforms and service providers. (PO-1,11, PSO-2)

**CO2:** Illustrate the significance of various types of virtualizations. (PO-1,2,3, PSO-2)

**CO3:** Identify the architecture, delivery models and industrial platforms for cloud computing based applications. (PO-1,2,3,11, PSO-2)

**CO4:** Analyze the role of security aspects in cloud computing. (PO 1,2,3,6,7, PSO-2)

**CO5:** Demonstrate cloud applications in various fields using suitable cloud platforms. (PO-1,2,3,11, PSO-2)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**CIE for the theory component of the IPCC (maximum marks 50)**

IPCC means practical portion integrated with the theory of the course.

CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.

25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90%

of the syllabus.

Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for 25 marks).

The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC.

**CIE for the practical component of the IPCC**

The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report.

15 marks for the conduction of the experiment and preparation of laboratory record, and 10 marks for the test to be conducted after the completion of all the laboratory sessions.

On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.

Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to 15 marks.

The laboratory test after completion of all the experiments shall be conducted for 50 marks and scaled down to 10 marks.

Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for 25 marks.

The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

**SEE for IPCC**

Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course (duration 03 hours)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

**The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.**

**Suggested Learning Resources:**

**Text Books:**


1. Kai Hwang, Geoffrey C Fox, and Jack J Dongarra, Distributed and Cloud Computing, Morgan Kaufmann, Elsevier 2012
2. Dan C. Marinescu, Cloud Computing Theory and Practice, Morgan Kaufmann, 2nd Edition, Elsevier 2018
3. Google Cloud Teaching Resources – LMS [for practical component]
4. AWS Cloud Developing – AWS Academy Courses [for practical component]


**Reference Books:**


1. Rajkumar Buyya, Christian Vecchiola, and Thamrai Selvi, Mastering Cloud Computing McGrawHill Education, 1st Edition, 2017
2. Toby Velte, Anthony Velte, Cloud Computing: A Practical Approach, McGraw-Hill Education, 2017.
3. George Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, O'Reilly Publication, 1st Edition, 2009
4. John Rhoton, Cloud Computing Explained: Implementation Handbook for Enterprises, Recursive Press, 2nd

Edition, 2009.

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

https://freevideolectures.com/course/4639/nptel-cloud-computing/1.
https://www.youtube.com/watch?v=RWgW-CgdIk0
https://www.javatpoint.com/cloud-service-provider-companies

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

Installation of virtualization software (Virtual box, Xen etc..) and run applications with different OS.
- 10 Marks

| ANALYSIS & DESIGN OF ALGORITHMS LAB | | | |
|---|---|---|---|
| **Course Code** | 25ISL45 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 0:0:2 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 28 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Practical | | |

**Prerequisites:**

The students should have knowledge on

- Fundamental of Data Structures and Programming Language
- Understanding of Algorithms
- Basic Mathematics
- Problem-Solving Skills

**Course Objectives:**

- **CLO1:** To design and implement various algorithms in C/C++ programming using suitable development tools to address different computational challenges.
- **CLO2:** To apply diverse design strategies for effective problem-solving.
- **CLO3:** To Measure and compare the performance of different algorithms to determine their efficiency and suitability for specific tasks.

| Sl.No | Experiments |
|---|---|
| 1 | Design and implement C/C++ Program to find Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm. |
| 2 | Design and implement C/C++ Program to find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm. |
| 3 | a. Design and implement C/C++ Program to solve All-Pairs Shortest Paths problem using Floyd's algorithm.<br>b. Design and implement C/C++ Program to find the transitive closure using Warshal's algorithm. |
| 4 | Design and implement C/C++ Program to find shortest paths from a given vertex in a weighted connected graph to other vertices using Dijkstra's algorithm. |
| 5 | Design and implement C/C++ Program to obtain the Topological ordering of vertices in a given digraph. |
| 6 | Design and implement C/C++ Program to solve 0/1 Knapsack problem using Dynamic Programming method. |
| 7 | Design and implement C/C++ Program to solve discrete Knapsack and continuous Knapsack problems using greedy approximation method. |
| 8 | Design and implement C/C++ Program to find a subset of a given set S = {sl , s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. |
| 9 | Design and implement C/C++ Program to sort a given set of n integer elements using Selection Sort method and compute its time complexity. Run the program for varied values of n> 5000 and record the time taken to sort. Plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator. |
| 10 | Design and implement C/C++ Program to sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of n> 5000 and record the time taken to sort. Plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator. |

| | |
|---|---|
| 11 | Design and implement C/C++ Program to sort a given set of n integer elements using Merge Sort method and compute its time complexity. Run the program for varied values of n> 5000, and record the time taken to sort. Plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator. |
| 12 | Design and implement C/C++ Program for N Queen's problem using Backtracking. |

**Course outcomes (Course Skill Set):**

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

**CO1:** Develop programs to solve computational problems using suitable algorithm design strategy. (PO-1,2,3,4.5, PSO-1,3)

**CO2:** Compare algorithm design strategies by developing equivalent programs and observing running times for analysis (Empirical). (PO-1,2,3,4,5. PSO-1,3)

**CO3:** Make use of suitable integrated development tools to develop programs. (PO-1,2,3,4,5, PSO-1,3)

**CO4:** Choose appropriate algorithm design techniques to develop solution to computational and complex problems. (PO-1,2,3,4,5, PSO-1,3)

**CO5:** Demonstrate and present the development of program, its execution and running time(s) and record the results/inferences. (PO-1,2,3,4,5, PSO-1,3)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- CIE marks for the practical course are 50 Marks. The split-up of CIE marks for record/ journal and test are in the ratio 60:40.
- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to 20 marks (40% of the maximum marks). The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student
- Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**Semester-End Examination:**

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of

the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University

- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question from PART-A and one question from PART-B (experiment) from the questions lot prepared by the examiners jointly and 50% weightage for each part.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners) Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero. The minimum duration of SEE is 02 hours.

| LINEAR ALGEBRA | | | |
|---|---|---|---|
| **Course Code** | 25IS46A | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 3:2:0 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 42 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge of

- Basics of Vectors
- Basic Mathematics
- Problem-Solving Skills

**Course objectives:**

The objectives of the course are to facilitate the learners to:

**CLO1:** To equip the students with standard concepts and tools in Linear algebra which will find the useful in their disciplines.

**CLO2:** Gain the knowledge of linear algebra tools and concepts to implement them in their core domain.

**CLO3:** Improve their mathematical thinking and acquire skills required for sustained lifelong learning

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Module-1** **9 Hours**

**VECTOR SPACES:** Introduction, Vector spaces, Subspaces, Linear Combinations, Linear Spans, row space and column space of a Matrix, Linear Dependence and Independence, Basis and Dimension, Problems.

**Text Book:1**, 3.1,3.2,3.3,3.4 and 3.5

**Module-2** **9 Hours**

**LINEAR TRANSFORMATIONS:** Introduction, Linear Mappings, Geometric linear transformation of i2, Kernel and Image of a linear transformations, Rank-Nullity Theorem (No proof), Matrix representation of linear transformations, Singular and Non-singular linear transformations, Invertible linear transformations, Problems

**Text Book:1**, 8.1,8.2 and 8.3

**Module 3** **8 hours**

**EIGENVALUES AND EIGENVECTORS:** Introduction, Polynomials of Matrices, Applications of Cayley-Hamilton Theorem, Eigen spaces of a linear transformation, Characteristic and Minimal Polynomials of Block Matrices, Jordan Canonical form.

**Text Book:1,** 6.1,6.2,6.3,6.4 and 6.5

**Module-4** **8 Hours**

**INNER PRODUCT SPACES:** Inner products, inner product spaces, length and orthogonality, orthogonal sets and Bases, projections, Gram-Schmidt process, QR-factorization, least squares problem and least square error

**Text Book:1,** 3.4 **Text Book:2,** 6.7

**Module-5** **8 Hours**

**OPTIMIZATION TECHNIQUES IN LINEAR ALGEBRA:** Diagonalization and Orthogonal diagonalization of real symmetric matrices, quadratic forms and its classifications, Hessian Matrix, Method of steepest descent, Singular value decomposition. Dimensionality reduction – Principal component analysis.

**Text Book:2,** 9.1,9.2,9.3 and 9.4

**Course outcome (Course Skill Set)**

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

**CO1**. Explain the concepts of vector spaces, subspaces, bases, dimension and their properties (PO – 1,2,3 ) .
**CO2**. Use matrices and linear transformations to solve the given problem (PO – 1,2,3)
**CO3**. Compute Eigenvalues and Eigenvectors for the linear transformations (PO – 1,2,3)
**CO4**. Determine orthogonality of inner product spaces (PO – 1,2,3)
**CO5**. Apply the optimization techniques to solve the problems (PO – 1,2,3)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation(CIE):**

● For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment
Test component, there are 25 marks.

● The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

● Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.

● For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:** Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4.Marks scored shall be proportionally reduced to 50 marks

**Suggested Learning Resources:**

**Textbook:**

1. David C. Lay, Steven R. Lay, Judi J Mc. Donald: “Linear Algebra and its applications”, Pearson Education, 6th Edition, 2021.
2. Gilbert Strang: “Linear Algebra and its applications”, Brooks Cole, 4th edition 2005.

**Reference Books:**

1 Richard Bronson & Gabriel B. Costa: “Linear Algebra: An Introduction”, 2nd edition. Academic Press, 2014.

2. Seymour Lipschutz, Marc Lipso: “Theory and problems of linear algebra”, Schaum’s outline series - 6th edition, 2017, McGraw-Hill Education.
3. Marc Peter Deisennroth, A. Aldo Faisal, Cheng Soon Ong: “Mathematics for Machine learning”, Cambridge University Press, 2020

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

- http://nptel.ac.in/courses.php?disciplineID=111
- 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
- Seminar

| DISCRETE MATHEMATICAL STRUCTURES | | | |
|---|---|---|---|
| **Course Code** | 25IS46B | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 3:2:0 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 42 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge on

- Basics of Logics
- Understanding Algebraic Properties
- Basic Mathematics
- Problem-Solving Skills

**Course Objectives:**

**CLO1:** To help students to understand discrete and continuous mathematical structures.
**CLO2:** To impart basics of relations and functions.
**CLO3:** To facilitate students in applying principles of Recurrence Relations to find the generating functions and solve the Recurrence relations.
**CLO4:** To have the knowledge of groups and their properties to understand the importance of algebraic properties relative to various number systems.

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Module-1** **9 Hours**

**Fundamentals of Logic:** Basic Connectives and Truth Tables, Logic Equivalence − The Laws of Logic, Logical Implication − Rules of Inference. The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems.

**Text Book:2**,2.1,2.2 and 2.3

**Module-2** **9 Hours**

**Properties of the Integers:** Mathematical Induction, The Well Ordering Principle − Mathematical Induction, Recursive Definitions.

**Fundamental Principles of Counting**: The Rules of Sum and Product, Permutations, Combinations − The Binomial Theorem, Combinations with Repetition.

**Text Book:2,** 1.1,1.2,1.3 ,1.4 ,4.1 and 4.2

**Module-3** **8 Hours**

**Relations and Functions:** Cartesian Products and Relations, Functions − Plain and One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions.

**Properties of Relations**: Computer Recognition − Zero-One Matrices and Directed Graphs, Partial Orders − Hasse Diagrams, Equivalence Relations and Partitions.

**Text Book:2**, 5.1,5.2,5.3,5.5 ,5.6,7.1,7.2,7.3 and 7.4

**Module-4** **8 Hours**

**The Principle of Inclusion and Exclusion:** The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements −Nothing is in its Right Place, Rook Polynomials.

**Recurrence Relations**: First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients.

**Text Book:2**, 8.1,8.2,8.3,8.4,10.1 and 10.2

**Module-5** **8 Hours**

**Introduction to Groups Theory:** Definitions and Examples of Particular Groups Klein 4-group, Additive group of Integers modulo n, Multiplicative group of Integers modulo-p and permutation groups, Properties of groups, Subgroups, cyclic groups, Cosets, Lagrange's Theorem.

**Text Book:1,** 7.1,7.2 and 7.3

## Course outcome (Course Skill Set)

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

**CO1.** Apply concepts of logical reasoning and mathematical proof techniques in proving theorems and statements **(**PO-1,2,3).

**CO2.** Demonstrate the application of discrete structures in different fields of computer science **(**PO- 1,2,3)

**CO3.** Apply the basic concepts of relations, functions and partially ordered sets for computer representations **(**PO-1,2,3)

**CO4.** Solve problems involving recurrence relations and generating functions **(**PO-1,2,3)

**CO5.** Illustrate the fundamental principles of Algebraic structures with the problems related to coputer science & engineering **(**PO- 1,2,3)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation(CIE):**

● For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.

● The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

● Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.

● For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:** Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.

2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4.Marks scored shall be proportionally reduced to 50 marks

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

- http://nptel.ac.in/courses.php?disciplineID=111
- http://www.class-central.com/subject/math(MOOCs)
- http://academicearth.org/
- VTU e-Shikshana Program
- VTU EDUSAT Program.
- **http://www.themathpage.com/**
- **http://www.abstractmath.org/**
- **http://www.ocw.mit.edu/courses/mathematics/**

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

- Quizzes
- Assignments
- Seminar

| GRAPH THEORY | | | |
|---|---|---|---|
| **Course Code** | 25IS46C | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 3:2:0 | **SEE Marks** | 50 |
| **Credits** | 03 | **Total Marks** | 100 |
| **Contact Hours** | 42 | **Exam Hours** | 3 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:**

The students should have knowledge on

- Basics of Graphs
- Basic Mathematics
- Problem-Solving Skills

**Course objectives:**

**CLO1:** Understand the basic concepts of graphs and their properties, and operations of graphs.
**CLO2:** Hamiltonian and Euler graphs, trees and matrix representation of the graph.
**CLO3:** Apply the concepts of a planar graph, matching and colouring in computer science engineering.

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**Module-1** **9 Hours**

**Introduction to Graphs:** Introduction- Basic definition − Application of graphs − finite, infinite and bipartite graphs − Incidence and Degree − Isolated vertex, pendant vertex and Null graph. Paths and circuits − Isomorphism, sub-graphs, walks, paths and circuits, connected graphs, disconnected graphs and components.
**Text Book:1**, 1.1 ,1.2,1.3,1.4,1.5,2.1,2.2,2.3,2.4 and 2.5 **Text Book:2,** 1.2

**Module-2** **9 Hours**

**Eulerian and Hamiltonian graphs:** Euler graphs, Operations on graphs, Hamiltonian paths and circuits, Travelling salesman problem. Directed graphs − types of digraphs, Digraphs and binary relation.
**Text Book:1**, 2.6,2.7,2.8,2.9,2.10,9.2 and 9.3

**Module-3** **8 Hours**

**Trees** − properties, pendant vertex, Distance and centres in a tree - Rooted and binary trees, counting trees, spanning trees.
**Connectivity Graphs**: Vertex Connectivity, Edge Connectivity, Cut set and Cut Vertices, Fundamental circuits.
**Text Book:1,** 3.1,3.2,3.3,3.4,3.5,3.6,3.7,4.1,4.2,4.3,4.4 and 4.5

**Module-4** **8 Hours**

**Planar Graphs:** Planar graphs, Kuratowski's theorem (proof not required), Different representations of planar graphs, Euler's theorem, Geometric dual.
**Graph Representations:** Matrix representation of graphs-Adjacency matrix, Incidence Matrix, Circuit Matrix, Path Matrix.
**Text Book:1**, 5.2,5.3,5.4,5.6,7.1,7.3,7.8 and 7.9

**Module-5** **8 Hours**

**Graph Colouring**: Colouring- Chromatic number, Chromatic polynomial, Matchings, Coverings, Four colour problem and Five colour problem. Greedy colouring algorithm
**Text Book:1,** 8.1,8.2,8.3,8.4,8.5 and 8.6 **Reference Book:3**, 5.2

**Course outcome (Course Skill Set)**

At the end of the course, the student will be able to:
**CO1**. Explain the fundamental concepts of properties and representation of graphs (PO – 1,2,3)
**CO2**. Solve the problems involving characterization and operations on graphs (PO – 1,2,3)
**CO3**. Apply concepts of trees and graph connectivity to solve real world problems (PO – 1,2,3)
**CO4**. Apply the concepts of planar graph and graph representations to solve the given problem (PO – 1,2,3)
**CO5**. Use the concepts of matching and coloring of graphs to solve the real world problems PO -1,2,3)

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

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation(CIE):**

● For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.

● The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

● Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.

● For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:** Theory SEE will be conducted as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4.Marks scored shall be proportionally reduced to 50 marks

**Suggested Learning Resources:**

**Textbook:**

1. Narsingh Deo, Graph theory with the applications to engineering & Computer Science, Dovers Publications, 2016
2. J.A. Bondy and U.S.R. Murty. Graph theory with Applications, Springer, 1st edition, 2008.

**Reference Books:**

1. Garry Chartand and Ping Zhang, Introduction to Graph Theory, Tata McGraw-Hill, 2006.
2. Frank Harary, Graph Theory, Narosa Publishing House, Latest edition.
3. R. Diestel, Graph Theory, free online edition, 2016: diestel-graph-theory.com/basic.html.
4. Douglas B. West, Introduction to Graph Theory, Prentice Hall India Ltd.,2001
5. Robin J. Wilson, Introduction to Graph Theory, Longman Group Ltd.,2010

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

- http://nptel.ac.in/courses.php?disciplineID=111
- 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
- Seminar

| UI/UX DESIGN TOOL | | | |
|---|---|---|---|
| **Course Code** | 25ISL47A | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T:P)** | 0:0:2 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 28 | **Exam Hours** | 2 |
| **Examination type (SEE)** | Practical | | |

**Prerequisites:**

The students should have knowledge of

- Familiarity with Digital Tools and Navigation
- Basic Knowledge of HTML/CSS
- Awareness of User-Centered Design

**Course Objectives:**

**CLO1:** Understand and differentiate core concepts of usability, user experience (UX), and user interface (UI) design.
**CLO2:** Conduct user research and develop personas to define user-centered design requirements.
**CLO3:** Apply design thinking and ideation techniques to create effective conceptual designs.
**CLO4:** Design and prototype low- to high-fidelity user interfaces using Figma.
**CLO5:** Evaluate and refine UX designs using metrics, usability testing, and integration with software development workflows.

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**LIST OF PROGRAMS**

1. Figma Interface Familiarization
   Create a sample artboard with text, shapes, and icons using the Figma workspace, tools, frames, layers, and properties.
2. Color Theory and Typography Practice
   Create a design sheet showcasing color palettes and a typography hierarchy (H1, H2, Body text).
3. Designing UI Components
   Create reusable buttons, input fields, radio buttons, and dropdown menus using components and variants.
4. Wireframing a Landing Page (Low Fidelity)
   Create a low-fidelity wireframe for a landing page using placeholders and basic layout techniques.
5. Building User Personas
   Design at least two user personas using frames, text, icons, and avatars in Figma.
6. User Journey Mapping
   Design a user journey map including touchpoints, pain points, and user actions for a selected app or service.
7. Storyboarding a User Scenario
   Create a 4–6 frame storyboard of a user performing a task in a digital product.
8. Creating a Mobile App Wireframe
   Design wireframes for a mobile app flow such as login → home → profile using Figma's auto layout feature.

9. Designing and Linking Interactive Prototypes
   Add navigation links, hover states, and transitions to screens to simulate a working prototype.
10. UI Redesign Challenge
   Redesign a poorly designed website or app screen and provide the improved version with justifications.
11. Usability Testing and Feedback Integration
   Share the prototype, collect feedback from peers or users, and iterate on the design based on usability principles.
12. Final Project – Complete App or Website UI
   Design and prototype a complete user interface flow for a selected application, including personas, wireframes, and interaction design.

**Course outcomes (Course Skill Set):**

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

**CO1:** Differentiate between usability, user experience (UX), and user interface (UI) design principles. (PO-1,9, PSO-1)

**CO2:** Conduct user research and develop personas to identify and define user-centered design requirements. (PO-2,5,9, PSO-1)

**CO3:** Use design thinking and ideation methods to create effective conceptual and interaction designs. (PO-1,3,5,9, PSO-1)

**CO4:** Develop low- to high-fidelity prototypes and UI layouts using Figma and related design tools.(PO-1,3,5,6, PSO-1,3)

**CO5:** Evaluate UX designs using usability principles, UX metrics, and integrate feedback into improvements. (PO-2,4,5,9, PSO-1)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- CIE marks for the practical course are 50 Marks. The split-up of CIE marks for record/ journal and test are in the ratio 60:40.
- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to 20 marks (40% of the maximum marks). The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student
- Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**Semester-End Examination:**

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question from PART-A and one question from PART-B (experiment) from the questions lot prepared by the examiners jointly and 50% weightage for each part.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners) Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero. The minimum duration of SEE is 02 hours.

**Suggested Learning Resources:**

**Text Books:**

1. UI Design with Figma: From Beginner to Pro by D-Libro (2025), A step-by-step guide covering fundamentals of wireframing, mockups, typography, prototyping, and real-world UI design workflows using Figma.

**Reference Book:**

1. Figma Handbook by Meng To (Design+Code) : A digital manual with 102 sections and ~6 hours of video content—covers everything from interface basics to advanced prototyping, plugins, responsive layouts, and component systems.

| DBMS WITH SQL AND MONGO DB | | | |
|---|---|---|---|
| **Course Code** | 25ISL47B | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 0:0:2 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 28 | **Exam Hours** | 2 |
| **Examination type (SEE)** | Practical | | |

**Prerequisites:**

The students should have knowledge on

- Basic Programming Skills
- Understanding of Algorithms
- Basic Mathematics
- Problem-Solving Skills

**Course Objectives:**

- **CLO1:** Practice relational (SQL) and NoSQL (MongoDB) database operations.
- **CLO2:** Develop skills in schema creation, data manipulation, and complex queries.
- **CLO3:** Work with real-world data examples.

**Teaching-Learning Process (General Instructions)**

1. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
2. Use of video/animation to explain the functioning of various concepts.
3. Encourage collaborative (group learning) in the class.
4. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
5. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
6. Use animations/videos to help students understand concepts.
7. Demonstrate concepts using a suitable programming language.

**PART-A**

1. Creating Tables, Users, and Constraints

Table: Employee(EMPNO, ENAME, JOB, MANAGER_NO, SAL, COMMISSION)

a) Create a new user and grant all permissions to the user.
b) Create the Employee table.
c) Insert three records into Employee table using INSERT.
d) Use ROLLBACK to undo the transaction.
e) Check the results after rollback.
f) Add PRIMARY KEY and NOT NULL constraints to the Employee table.
g) Try to insert NULL values and verify the constraint enforcement.

2. Altering Tables and DML Operations

Table: Employee(EMPNO, ENAME, JOB, MGR, SAL)

a) Add a new column COMMISSION with appropriate data type.
b) Insert five records into the Employee table.
c) Update JOB column details for any employee.
d) Rename a column in the Employee table using the ALTER command.
e) Delete the employee whose EMPNO is 105.

3. Queries using Aggregate Functions

Table: Employee(E_id, E_name, Age, Salary)

a. Create the Employee table with specified attributes.
b. Count the number of employees in the table.
c. Find the maximum age of employees.

d. Find the minimum age of employees.
e. Display salaries in ascending order.
f. Find grouped salaries using GROUP BY.

4. Triggers
Table: CUSTOMERS(ID, NAME, AGE, ADDRESS, SALARY)
Create a row-level trigger for the CUSTOMERS table that fires on INSERT, UPDATE, or DELETE operations and displays the salary difference between the old and new SALARY.

5. Cursors
Table: Employee(E_id, E_name, Age, Salary)
Create a cursor for the Employee table:

- Declare variables
- Open the cursor
- Extract values from the table using the cursor
- Close the cursor

**PART-B**

6. Basic CRUD Operations in MongoDB
a. Execute basic commands and operations in MongoDB:

- Insert documents
- Query documents
- Update documents
- Delete documents
- Apply Projection to select specific fields

Note: Use any sample collection
Reference: Book 1, Chapter 4
b. Illustrate the use of WHERE, AND, and OR operators in MongoDB queries.

7. Selecting and Limiting Fields
a. Develop a query to select specific fields and exclude other fields from any collection.
b. Use limit and find to display only the first 5 documents from the results of (a).
References: Book 1, Chapter 4; Book 2, Chapter 5

8. Using Query Selectors
a. Execute queries using comparison and logical selectors; list the results from any collection.
b. Execute queries using geospatial and bitwise selectors; list the results from any collection.
Reference: Book 3, Chapter 13

9. Using Projection Operators
Create and demonstrate the use of projection operators:

- $ (Positional Operator)
- $elemMatch
- $slice

Reference: Book 3, Chapter 14
10. Aggregation Operators
Execute aggregation operations using the following operators:

- $avg
- $min
- $max
- $push
- $addToSet

Students should run multiple queries to demonstrate the various aggregation operators.

Reference: Book 3, Chapter 15

11. Aggregation Pipeline

Develop an aggregation pipeline that includes the following stages:

- $match
- $group
- $sort
- $project
- $skip

Students should demonstrate multiple queries using these pipeline operators.

Reference: Book 2, Chapter 6

12. Real-World Query Examples

a. Find all listings with listing_url, name, address, and host_picture_url in the Listings And Reviews collection where the host has a picture URL.

b. Using an E-commerce collection, write a query to display the reviews summary.

Reference: Book 2, Chapter 6

13. Text Search

a. Develop a query to perform text search on a Catalog data collection for a given word.

b. Develop queries to exclude documents containing certain words or phrases.

Reference: Book 2, Chapter 9

14. Aggregation Pipeline for Text Search

Develop an aggregation pipeline to demonstrate text search on a Catalog data collection.

Reference: Book 2, Chapter 9

**Course outcomes (Course Skill Set):**

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

**CO1:** Design, implement, and query relational schemas. (PO-1,2,3,5,11, PSO-1,3)

**CO2:** Perform CRUD and aggregation with MongoDB. (PO-1,2,3,5,11, PSO-1,3)

**CO3:** Demonstrate proficiency with SQL joins, views, indexing, and transactions. (PO-1,2,3,5,11, PSO-1,3)

**CO4:** Apply database concepts in small applications. (PO-1,2,3,5,11, PSO-1,3)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- CIE marks for the practical course are 50 Marks. The split-up of CIE marks for record/ journal and test are in the ratio 60:40.
- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.

- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to 20 marks (40% of the maximum marks). The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student
- Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**Semester-End Examination:**

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question from PART-A and one question from PART-B (experiment) from the questions lot prepared by the examiners jointly and 50% weightage for each part.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners) Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero. The minimum duration of SEE is 02 hours.

**Suggested Learning Resources :**

**Reference Books:**


1. MongoDB: The Definitive Guide — Kristina Chodorow, 2nd Edition, O'Reilly, 2013.
2. MongoDB in Action — Kyle Banker et al., 2nd Edition, Manning Publications, 2016.
3. MongoDB Complete Guide — Manu Sharma, 1st Edition, BPB Publications, 2023.


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

- Installation of MongoDB: https://www.youtube.com/watch?v=dEm2AS5amyA
- Aggregation in MongoDB: https://www.youtube.com/watch?v=vx1C8EyTa7Y
- Additional Resources:
- MongoDB in Action — Source Code Download: https://www.manning.com/downloads/529
- MongoDB Practice Exercises: https://www.w3resource.com/mongodb-exercises/

| TECHNICAL WRITING USING LATEX | | | |
|---|---|---|---|
| **Course Code** | 25ISL47C | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T:P)** | 0:0:2 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 28 | **Exam Hours** | 2 |
| **Examination type (SEE)** | Practical | | |

**Prerequisites:**

The students should have knowledge of

- Knowledge on Familiarity with Document Preparation
- Knowledge on Typing and Editing Skills
- Knowledge on Basic Understanding of Mathematical Notation
- Knowledge on Logical and Structured Thinking

**Course Objectives:**

**CLO1:** To introduce the basic syntax and semantics of the LaTeX scripting language
**CLO2:** To understand the presentation of tables and figures in the document
**CLO3:**To illustrate the LaTeX syntax to represent the theorems and mathematical equations
**CLO4:** To make use of the libraries (Tikz, algorithm) to design the diagram and algorithms in the document

**Teaching-Learning Process (General Instructions)**

8. Lecture method (L) need not be only a traditional lecture method; alternative effective teaching methods may be adopted to attain the outcomes.
9. Use of video/animation to explain the functioning of various concepts.
10. Encourage collaborative (group learning) in the class.
11. Ask at least three HOT (Higher Order Thinking) questions in the class to promote critical thinking.
12. Adopt Problem Based Learning (PBL) to foster students' analytical and design thinking skills such as designing, evaluating, generalizing, and analyzing information rather than simple recall.
13. Use animations/videos to help students understand concepts.
14. Demonstrate concepts using a suitable programming language.

**LIST OF PROGRAMS**

1. Develop a LaTeX script to create a simple document that consists of 2 sections [Section1, Section2], and a paragraph with dummy text in each section. And also include header [title of document] and footer [institute name, page number] in the document.
2. Develop a LaTeX script to create a document that displays the sample Abstract/Summary
3. Develop a LaTeX script to create a simple title page of the VTU project Report [Use suitable Logos and text formatting]
4. Develop a LaTeX script to create the Certificate Page of the Report [Use suitable commands to leave the blank spaces for user entry]
5. Develop a LaTeX script to create a document that contains the following table with proper labels.

| SNO | **USN** | **Student Name** | **Marks** | | |
|---|---|---|---|---|---|
| | | | **Subject1** | **Subject2** | **Subject3** |
| 1 | 4XX22XX001 | Name1 | 89 | 60 | 90 |
| 2 | 4XX22XX002 | Name2 | 78 | 45 | 98 |
| 3 | 4XX22XX003 | Name3 | 67 | 55 | 59 |

6. Develop a LaTeX script to include the side-by-side graphics/pictures/figures in the document by using the subgraph concept

7. Develop a LaTeX script to create a document that consists of the following two mathematical equations

$$x = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$$

$$= \frac{-2 \pm \sqrt{2^2-4*(1)*(-8)}}{2*1}$$

$$= \frac{-2 \pm \sqrt{4+32}}{2}$$

$$\varphi_\sigma^\lambda A_t = \sum_{\pi \in C_t} \operatorname{sgn}(\pi) \varphi_\sigma^\lambda \varphi_\pi^\lambda$$

$$= \sum_{\tau \in C_{\sigma t}} \operatorname{sgn}(\sigma^{-1}\tau\sigma) \varphi_\sigma^\lambda \varphi_{\sigma^{-1}\tau\sigma}^\lambda$$

$$= A_{\sigma t} \varphi_\sigma^\lambda$$

8. Develop a LaTeX script to demonstrate the presentation of Numbered theorems, definitions, corollaries, and lemmas in the document
9. Develop a LaTeX script to create a document that consists of two paragraph s with a minimum of 10 citations in it and display the reference in the section
10. Develop a LaTeX script to design a simple tree diagram or hierarchical structure in the document with appropriate labels using the Tikz library
11. Develop a LaTeX script to present an algorithm in the document using algorithm /algorithmic /algorithm 2e library
12. Develop a LaTeX script to create a simple report and article by using suitable commands and formats of user choice.

**Course outcomes (Course Skill Set):**

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

**CO1:** Apply basic LaTeX command to develop simple documents. (PO-1,2,3,5, PSO -1,3)

**CO2:** Develop LaTeX script to present the tables and figures in the document. (PO-1,2,3,5, PSO -1,3)

**CO3**: Illustrate LaTeX script to present theorems and mathematical equations in the document. (PO-1,2,3,5, PSO-1,3)

**CO4**: Develop programs to generate the complete report with citations and a bibliography. (PO- 1,2,3,5, PSO-1,3)

**CO5:** Illustrate the use of Tikz and algorithm libraries to design graphics and algorithms in the document. (PO -1,2,3,5, PSO -1,3)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- CIE marks for the practical course are 50 Marks. The split-up of CIE marks for record/ journal and test are in the ratio 60:40.
- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).

- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to 20 marks (40% of the maximum marks). The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student
- Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**Semester-End Examination:**

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question from PART-A and one question from PART-B (experiment) from the questions lot prepared by the examiners jointly and 50% weightage for each part.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners) Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero. The minimum duration of SEE is 02 hours.

**Suggested Learning Resources:**

**Text Books:**


1. A Short Introduction to LaTeX by FIRUZAKARMALI (AIBARA), A book for beginners,2019
2. Formatting Information: A Beginner's Introduction to Type setting with LaTeX, BY PETERFLYNN, Comprehensive TeX Archive Network (2005)


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

- LaTeX TUTORIAL: [https://latex-tutorial.com/tutorials/]
- LaTeX TUTORIAL: [https://www.javatpoint.com/latex]

| Universal Human Values | | | |
|---|---|---|---|
| **Course Code** | 25UH48 | **CIE Marks** | 50 |
| **Teaching Hours/Week (L: T: P)** | 1:0:0 | **SEE Marks** | 50 |
| **Credits** | 01 | **Total Marks** | 100 |
| **Contact Hours** | 14:0:0 | **Exam Hours** | 03 |
| **Examination type (SEE)** | Theory | | |

**Prerequisites:** NIL

**Course Objectives:**

This course is intended to:

**CLO1:** To help the students appreciate the essential complementarity between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings.

**CLO2:**To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence. Such a holistic perspective forms the basis of Universal Human Values and movement towards value-based living in a natural way.

**CLO3:**To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature.

**CLO4:**This course is intended to provide a much-needed orientation input in value education to the young enquiring minds.

**Teaching-Learning Process (General Instructions):**

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

1. The methodology of this course is explorational and thus universally adaptable. It involves a systematic and rational study of the human being vis-à-vis the rest of existence.
2. 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 skills.
3. State the need for UHV activities and its present relevance in the society and provide real-life examples.
4. Support and guide the students for self-study activities.
5. You will also be responsible for assigning homework, grading assignments and quizzes, and documenting students' progress in real activities in the field.
6. This process of self-exploration takes the form of a dialogue between the teacher and the students to begin with, and then to continue within the student in every activity, leading to continuous self-evolution.
7. Encourage the students for group work to improve their creative and analytical skills.

**Module- 1** **2 Hours**

**Introduction to Value Education:**

Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education) Understanding Value Education, Self-exploration as the Process for Value Education, Continuous Happiness and Prosperity – the Basic Human Aspirations, Happiness and Prosperity – Current Scenario, Method to Fulfil the Basic Human Aspirations

**Module- 2** **3 Hours**

**Harmony in the Human Being :**

Understanding Human being as the Co-existence of the Self and the Body, Distinguishing between the Needs of the Self and the Body, The Body as an Instrument of the Self, Understanding Harmony in the Self, Harmony of the Self with the Body, Programme to ensure self-regulation and Health

| Module- 3 | 3 Hours |
|---|---|

**Harmony in the Family and Society :**

Harmony in the Family – the Basic Unit of Human Interaction, 'Trust' – the Foundational Value in Relationship, 'Respect' – as the Right Evaluation, Other Feelings, Justice in Human-toHuman Relationship, Understanding Harmony in the Society, Vision for the Universal Human Order

| Module- 4 | 3 Hours |
|---|---|

**Harmony in the Nature/Existence :**

Understanding Harmony in the Nature, Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature, Realizing Existence as Co-existence at All Levels, The Holistic Perception of Harmony in Existence

| Module- 5 | 3 Hours |
|---|---|

**Implications of the Holistic Understanding – a Look at Professional Ethics** :

Natural Acceptance of Human Values, Definitiveness of (Ethical) Human Conduct, A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order, Competence in Professional Ethics Holistic Technologies, Production Systems and Management Models-Typical Case Studies, Strategies for Transition towards Value-based Life and Profession

**Course outcomes (Course Skill Set):**

At the end of the course, students are expected to become more aware of themselves, and their surroundings (family, society, nature):

**CO1:** Apprehend the need of Value Education over Human aspirations ( PO-6, PO-7)

**CO2:** Assimilate Harmony over the physical needs and to overcome the self- needs for a prosperous life. ( PO-6, PO-7)

**CO3:** Recognize the need of Harmony in the Family and Society for a better World. ( PO-6, PO-7)

**CO4:** Explain the need of mutual understanding for Holistic Harmony in all the Levels of Human Existence. ( PO-6, PO-7)

**CO5:** Explain the Holistic understanding of Harmony and Professional Ethics at Individual Level and Society. (PO-6, PO-7)

**Assessment Details (both CIE and SEE):**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). The student is declared as a pass in the course if he/she secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

**Continuous Internal Evaluation:**

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered.
- Any two assignment methods, if an assignment is project-based, then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

**The sum of two tests, two assignments, will be out of 100 marks and will be scaled down to 50 marks**

**Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

**Semester-End Examination:**

SEE paper shall be set for 50 questions, each of the 01 marks. The pattern of the question paper is MCQ (multiple choice questions). The time allotted for SEE is 01 hour. The student has to secure a minimum of 35% of the maximum marks meant for SEE.

**Suggested Learning Resources:**

**Text Books:**


1. "The Textbook: A Foundation Course in Human Values and Professional Ethics", R R Gaur, R Asthana, G P Bagaria, 2nd Revised Edition, Excel Books, New Delhi, 2010. ISBN 978- 8-174- 46781-2
2. The Teacher"s Manual for A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G


**Reference Books:**


1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amar kantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj – Pandit Sunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)
14. Sussan George, 1976, How the Other Half Dies, Penguin Press. Reprinted 1986, 1991
15. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, Limits to Growth – Club of Rome's report, Universe Books.
16. A Nagraj, 1998, Jeevan Vidya Ek Parichay, Divya Path Sansthan, Amarkantak.
17. P L Dhar, RR Gaur, 1990, Science and Humanism, Commonwealth Publishers.
18. A N Tripathy, 2003, Human Values, New Age International Publishers.
19. SubhasPalekar, 2000, How to practice Natural Farming, Pracheen (Vaidik) KrishiTantraShodh, Amravati.
20. E G Seebauer & Robert L. Berry, 2000, Fundamentals of Ethics for Scientists & Engineers , Oxford University Press
21. M Govindrajran, S Natrajan & V.S. Senthil Kumar, Engineering Ethics (including Human Values), Eastern Economy Edition, Prentice Hall of India Ltd.
22. B P Banerjee, 2005, Foundations of Ethics and Management, Excel Books.
23. B L Bajpai, 2004, Indian Ethos and Modern Management, New Royal Book Co., Lucknow. Reprinted 2008.


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

1. https://www.youtube.com/channel/UCQxWr5QB_eZUnwxSwxXEkQw
2. https://www.youtube.com/watch?v=P4vjfE-YnVk&list=PLWDeKF97v9SP7wSlapZcQRrT7OH0ZlGC4
3. **Course handouts:**
   https://drive.google.com/drive/folders/1zioX_4L2fCNX4Agw282PN86pcZZT3 Osr?usp=sharing
4. **Presentation slides:**
   https://drive.google.com/drive/folders/1rMUKh1s0HPRBlpp_b1mpS-duNRcwS6YH?usp=sharing