B. E. COMMON TO ALL PROGRAMMES
Choice Based Credit System (CBCS) and Outcome Based Education (OBE)
SEMESTER - III

TRANSFORM CALCULUS, FOURIER SERIES AND NUMERICAL TECHNIQUES

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18MAT31</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teaching Hours/Week (L: T:P)</td>
<td>(2:2:0)</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Credits</td>
<td>03</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

Course Learning Objectives:
- To have an insight into Fourier series, Fourier transforms, Laplace transforms, Difference equations and Z-transforms.
- To develop the proficiency in variational calculus and solving ODE’s arising in engineering applications, using numerical methods.

Module-1

Laplace Transform: Definition and Laplace transforms of elementary functions (statements only). Laplace transforms of Periodic functions (statement only) and unit-step function – problems.

Inverse Laplace Transform: Definition and problems, Convolution theorem to find the inverse Laplace transforms (without Proof) and problems. Solution of linear differential equations using Laplace transforms.

Module-2

Fourier Series: Periodic functions, Dirichlet’s condition. Fourier series of periodic functions period \( \frac{2\pi}{2} \) and arbitrary period. Half range Fourier series. Practical harmonic analysis.

Module-3


Difference Equations and Z-Transforms: Difference equations, basic definition, z-transform-definition, Standard z-transforms, Damping and shifting rules, initial value and final value theorems (without proof) and problems, Inverse z-transform and applications to solve difference equations.

Module-4

Numerical Solutions of Ordinary Differential Equations(ODE’s):
Numerical solution of ODE’s of first order and first degree- Taylor’s series method, Modified Euler’s method. Runge-Kutta method of fourth order, Milne’s and Adam-Bash forth predictor and corrector method (No derivations of formulae)-Problems.

Module-5

Numerical Solution of Second Order ODE’s: Runge-Kutta method and Milne’s predictor and corrector method. (No derivations of formulae).

Calculus of Variations: Variation of function and functional, variational problems, Euler’s equation, Geodesics, hanging chain, problems.

Course outcomes: At the end of the course the student will be able to:
- CO1: Use Laplace transform and inverse Laplace transform in solving differential/integral equation arising in network analysis, control systems and other fields of engineering.
- CO2: Demonstrate Fourier series to study the behaviour of periodic functions and their applications in system communications, digital signal processing and field theory.
- CO3: Make use of Fourier transform and Z-transform to illustrate discrete/continuous function arising in wave and heat propagation, signals and systems.
- CO4: Solve first and second order ordinary differential equations arising in engineering problems using single step and multistep numerical methods.
- CO5: Determine the externals of functionals using calculus of variations and solve problems arising in dynamics of rigid bodies and vibrational analysis.

Question paper pattern:
- The question paper will have ten full questions carrying equal marks.
- Each full question will be for 20 marks.
- There will be two full questions (with a maximum of four sub-questions) from each module.
- Each full question will have sub-question covering all the topics under a module.
- The students will have to answer five full questions, selecting one full question from each module.

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Title of the Book</th>
<th>Name of the Author/s</th>
<th>Name of the Publisher</th>
<th>Edition and Year</th>
</tr>
</thead>
</table>

**Reference Books**

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Title of the Book</th>
<th>Name of the Author/s</th>
<th>Name of the Publisher</th>
<th>Edition and Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Advanced Engineering Mathematics</td>
<td>Chandrika Prasad and Reena Garg</td>
<td>Khanna Publishing,</td>
<td>2018</td>
</tr>
</tbody>
</table>

**Web links and Video Lectures:**
2. http://www.class-central.com/subject/math(MOOCs)
4. VTU EDUSAT PROGRAMME - 20
## Course Learning Objectives:

This course (18CS32) will enable students to:

- Explain fundamentals of data structures and their applications essential for programming/problem solving.
- Illustrate linear representation of data structures: Stack, Queues, Lists, Trees and Graphs.
- Demonstrate sorting and searching algorithms.
- Find suitable data structure during application development/Problem Solving.

### Module 1

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Array Operations:</strong> Traversing, inserting, deleting, searching, and sorting. Multidimensional Arrays, Polynomials and Sparse Matrices.</td>
</tr>
<tr>
<td><strong>Strings:</strong> Basic Terminology, Storing, Operations and Pattern Matching algorithms. Programming Examples.</td>
</tr>
<tr>
<td><strong>Textbook 1:</strong> Chapter 1: 1.2, Chapter 2: 2.2 - 2.7 Textbook 2: Chapter 1: 1.1 - 1.4, Chapter 3: 3.1 - 3.3, 3.5, 3.7, Chapter 4: 4.1 - 4.9, 4.14 Reference 3: Chapter 1: 1.4</td>
</tr>
</tbody>
</table>

### Module 2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Stacks:</strong> Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression.</td>
</tr>
<tr>
<td><strong>Recursion</strong> - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi, Ackerman's function.</td>
</tr>
<tr>
<td><strong>Queues:</strong> Definition, Array Representation, Queue Operations, Circular Queues, Circular queues using Dynamic arrays, Dequeues, Priority Queues, A Mazing Problem. Multiple Stacks and Queues. Programming Examples.</td>
</tr>
<tr>
<td><strong>Textbook 1:</strong> Chapter 3: 3.1 -3.7 Textbook 2: Chapter 6: 6.1 -6.3, 6.5, 6.7-6.10, 6.12, 6.13</td>
</tr>
</tbody>
</table>

### Module 3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Linked Lists:</strong> Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. Applications of Linked lists – Polynomials, Sparse matrix representation. Programming Examples</td>
</tr>
<tr>
<td><strong>Textbook 1:</strong> Chapter 4: 4.1 – 4.6, 4.8, Textbook 2: Chapter 5: 5.1 – 5.10, RBT: L1, L2, L3</td>
</tr>
</tbody>
</table>

### Module 4

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Trees:</strong> Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression, Programming Examples</td>
</tr>
</tbody>
</table>
### Module 5

**Graphs:** Definitions, Terminologies, Matrix and Adjacency List Representation Of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search.

**Sorting and Searching:** Insertion Sort, Radix sort, Address Calculation Sort.

**Hashing:** Hash Table organizations, Hashing Functions, Static and Dynamic Hashing.

**Files and Their Organization:** Data Hierarchy, File Attributes, Text Files and Binary Files, Basic File Operations, File Organizations and Indexing

**Textbook 1:** Chapter 6: 6.1–6.2, Chapter 7: 7.2, Chapter 8: 8.1–8.3

**Textbook 2:** Chapter 8: 8.1–8.7, Chapter 9: 9.1–9.3, 9.7, 9.9

**Reference 2:** Chapter 16: 16.1–16.7

### Course Outcomes:

- Use different types of data structures, operations and algorithms
- Apply searching and sorting operations on files
- Use stack, Queue, Lists, Trees and Graphs in problem solving
- Implement all data structures in a high-level language for problem solving.

### Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:


### Reference Books:

## Course Learning Objectives:
This course (18CS33) will enable students to:

- Explain the use of photoelectronics devices, 555 timer IC, Regulator ICs and uA741 opamp IC
- Make use of simplifying techniques in the design of combinational circuits.
- Illustrate combinational and sequential digital circuits
- Demonstrate the use of flipflops and apply for registers
- Design and test counters, Analog-to-Digital and Digital-to-Analog conversion techniques.

### Module 1


**Text Book 1 :** Part A: Chapter 2 (Section 2.9, 2.10, 2.11), Chapter 4 (Section 4.2, 4.3, 4.4), Chapter 7 (section 7.2, 7.3, 7.4, 7.6 to 7.11), Chapter 8 (section 8.1, 8.5), Chapter 9

**RBT: L1, L2**

### Module 2

Karnaugh maps: minimum forms of switching functions, two and three variable Karnaugh maps, four variable karnaugh maps, determination of minimum expressions using essential prime implicants, Quine-McClusky Method: determination of prime implicants, The prime implicant chart, petricks method, simplification of incompletely specified functions, simplification using map-entered variables

**Text Book 1: Part B: Chapter 5 (Sections 5.1 to 5.4) Chapter 6 (Sections 6.1 to 6.5)**

**RBT: L1, L2**

### Module 3

Combinational circuit design and simulation using gates: Review of Combinational circuit design, design of circuits with limited Gate Fan-in, Gate delays and Timing diagrams, Hazards in combinational Logic, simulation and testing of logic circuits

Multiplexers, Decoders and Programmable Logic Devices: Multiplexers, three state buffers, decoders and encoders, Programmable Logic devices, Programmable Logic Arrays, Programmable Array Logic.

**Text Book 1: Part B: Chapter 8, Chapter 9 (Sections 9.1 to 9.6)**

**RBT: L1, L2**

### Module 4

Introduction to VHDL: VHDL description of combinational circuits, VHDL Models for multiplexers, VHDL Modules.

Latches and Flip-Flops: Set Reset Latch, Gated Latches, Edge-Triggered D Flip Flop, 3,SR Flip Flop, J K Flip Flop, T Flip Flop, Flip Flop with additional inputs, Asynchronous Sequential Circuits

**Text Book 1: Part B: Chapter 10 (Sections 10.1 to 10.3), Chapter 11 (Sections 11.1 to 11.9)**

**RBT: L1, L2**
## Module 5

Registers and Counters: Registers and Register Transfers, Parallel Adder with accumulator, shift registers, design of Binary counters, counters for other sequences, counter design using SR and J K Flip Flops, sequential parity checker, state tables and graphs

**Text book 1:** Part B: Chapter 12(Sections 12.1 to 12.5), Chapter 13(Sections 13.1, 13.3)

RBT: L1, L2

### Course Outcomes:

The student will be able to:

- Design and analyze application of analog circuits using photo devices, timer IC, power supply and regulator IC and op-amp.
- Explain the basic principles of A/D and D/A conversion circuits and develop the same.
- Simplify digital circuits using Karnaugh Map, and Quine-McClusky Methods
- Explain Gates and flip flops and make us in designing different data processing circuits, registers and counters and compare the types.
- Develop simple HDL programs

### Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:

1. Charles H Roth and Larry L. Kinney, Analog and Digital Electronics, Cengage Learning, 2019

### Reference Books:

**Course Code**: 18CS34  
**CIE Marks**: 40  
**Number of Contact Hours/Week**: 3:0:0  
**SEE Marks**: 60  
**Total Number of Contact Hours**: 40  
**Exam Hours**: 03

**Course Learning Objectives**: This course (18CS34) will enable students to:

- Explain the basic sub systems of a computer, their organization, structure and operation.
- Illustrate the concept of programs as sequences of machine instructions.
- Demonstrate different ways of communicating with I/O devices and standard I/O interfaces.
- Describe memory hierarchy and concept of virtual memory.
- Describe arithmetic and logical operations with integer and floating-point operands.
- Illustrate organization of a simple processor, pipelined processor and other computing systems.

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>
| **Basic Structure of Computers**: Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement.  
**Machine Instructions and Programs**: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions.  
**Text book 1**: Chapter1 – 1.3, 1.4, 1.6 (1.6.1-1.6.4, 1.6.7), Chapter2 – 2.2 to 2.10  
**RBT**: L1, L2, L3 | 08 |

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>
| **Input/Output Organization**: Accessing I/O Devices, Interrupts – Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB.  
**Text book 1**: Chapter4 – 4.1, 4.2, 4.4, 4.5, 4.6, 4.7  
**RBT**: L1, L2, L3 | 08 |

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>
| **Memory System**: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations.  
**Text book 1**: Chapter5 – 5.1 to 5.4, 5.5 (5.5.1, 5.5.2), 5.6  
**RBT**: L1, L2, L3 | 08 |

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>
**Text book 1**: Chapter2-2.1, Chapter6 – 6.1 to 6.6  
**RBT**: L1, L2, L3 | 08 |

<table>
<thead>
<tr>
<th>Module 5</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>
| **Basic Processing Unit**: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro programmed Control.  
**Pipelining**: Basic concepts of pipelining,  
**Text book 1**: Chapter7, Chapter8 – 8.1  
**RBT**: L1, L2, L3 | 08 |

**Course Outcomes**: The student will be able to:

- Explain the basic organization of a computer system.
- Demonstrate functioning of different sub systems, such as processor, input/output, and memory.
- Illustrate hardwired control and micro programmed control, pipelining, embedded and other computing systems.
- Design and analyse simple arithmetic and logical units.

**Question Paper Pattern:**
- The question paper will have ten questions.
- Each full question consisting of 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**

**Reference Books:**
## SOFTWARE ENGINEERING

(Effective from the academic year 2018 -2019)

**SEMESTER – III**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS35</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

**Course Code 18CS35 CIE Marks 40**

**Number of Contact Hours/Week**

| Total Number of Contact Hours | 40 |

| Exam Hours | 03 |

**CREDITS –3**

**Course Learning Objectives:** This course (18CS35) will enable students to:

- Outline software engineering principles and activities involved in building large software programs. Identify ethical and professional issues and explain why they are of concern to software engineers.
- Explain the fundamentals of object oriented concepts.
- Describe the process of requirements gathering, requirements classification, requirements specification and requirements validation. Differentiate system models, use UML diagrams and apply design patterns.
- Discuss the distinctions between validation testing and defect testing.
- Recognize the importance of software maintenance and describe the intricacies involved in software evolution. Apply estimation techniques, schedule project activities and compute pricing.
- Identify software quality parameters and quantify software using measurements and metrics. List software quality standards and outline the practices involved.

### Module 1

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>


**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).

**RBT:** L1, L2, L3

### Module 2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

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:** Ch 1,2,3.

**RBT:** L1, L2, L3

### Module 3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

**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).

**RBT:** L1, L2, L3

### Module 4
<table>
<thead>
<tr>
<th><strong>Module 5</strong></th>
<th>08</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Project Planning:</strong> Software pricing (Sec 23.1). Plan-driven development (Sec 23.2). Project scheduling (Sec 23.3): Estimation techniques (Sec 23.5). <strong>Quality management:</strong> Software quality (Sec 24.1). Reviews and inspections (Sec 23.5). Software measurement and metrics (Sec 24.3). Software standards (Sec 24.2)</td>
<td>08</td>
</tr>
</tbody>
</table>

| **Course Outcomes:** The student will be able to: |
|---|---|
| • Design a software system, component, or process to meet desired needs within realistic constraints. |
| • Assess professional and ethical responsibility |
| • Function on multi-disciplinary teams |
| • Use the techniques, skills, and modern engineering tools necessary for engineering practice |
| • Analyze, design, implement, verify, validate, implement, apply, and maintain software systems or parts of software systems |

| **Question Paper Pattern:** |
|---|---|
| • The question paper will have ten questions. |
| • Each full question consisting of 20 marks |
| • There will be 2 full questions (with a maximum of four sub questions) from each module. |
| • Each full question will have sub questions covering all the topics under a module. |
| • The students will have to answer 5 full questions, selecting one full question from each module. |

| **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) |

<p>| <strong>Reference Books:</strong> |
|---|---|
| 2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India |</p>
<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS36</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –3**

**Course Learning Objectives:** This course (18CS36) will enable students to:

- Provide theoretical foundations of computer science to perceive other courses in the programme.
- Illustrate applications of discrete structures: logic, relations, functions, set theory and counting.
- Describe different mathematical proof techniques,
- Illustrate the importance of graph theory in computer science

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fundamentals of Logic</strong>: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference. Fundamentals of Logic contd.: The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems.</td>
<td>08</td>
</tr>
<tr>
<td><strong>Text book 1</strong>: Chapter2</td>
<td></td>
</tr>
<tr>
<td><strong>RBT</strong>: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Properties of the Integers</strong>: The Well Ordering Principle – Mathematical Induction, <strong>Fundamental Principles of Counting</strong>: The Rules of Sum and Product, Permutations, Combinations – The Binomial Theorem, Combinations with Repetition.</td>
<td>08</td>
</tr>
<tr>
<td><strong>Text book 1</strong>: Chapter4 – 4.1, Chapter1</td>
<td></td>
</tr>
<tr>
<td><strong>RBT</strong>: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Relations and Functions</strong>: Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions.</td>
<td>08</td>
</tr>
<tr>
<td><strong>Text book 1</strong>: Chapter5, Chapter7 – 7.1 to 7.4</td>
<td></td>
</tr>
<tr>
<td><strong>RBT</strong>: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The Principle of Inclusion and Exclusion</strong>: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials.</td>
<td>08</td>
</tr>
<tr>
<td><strong>Recurrence Relations</strong>: First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients.</td>
<td></td>
</tr>
<tr>
<td><strong>Text book 1</strong>: Chapter8 – 8.1 to 8.4, Chapter10 – 10.1, 10.2</td>
<td></td>
</tr>
<tr>
<td><strong>RBT</strong>: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 5</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Introduction to Graph Theory</strong>: Definitions and Examples, Sub graphs, Complements, and Graph Isomorphism,</td>
<td>08</td>
</tr>
<tr>
<td><strong>Trees</strong>: Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes</td>
<td></td>
</tr>
<tr>
<td><strong>Text book 1</strong>: Chapter11 – 11.1 to 11.2 Chapter12 – 12.1 to 12.4</td>
<td></td>
</tr>
<tr>
<td><strong>RBT</strong>: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

**Course Outcomes:** The student will be able to:

- Use propositional and predicate logic in knowledge representation and truth verification.
- Demonstrate the application of discrete structures in different fields of computer science.
- Solve problems using recurrence relations and generating functions.
- Application of different mathematical proofs techniques in proving theorems in the courses.
- Compare graphs, trees and their applications.

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**

ANALOG AND DIGITAL ELECTRONICS LABORATORY  
(Effective from the academic year 2018 -2019)  
SEMESTER – III

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CSL37</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>0:2:2</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Lab Contact Hours</td>
<td>36</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
<tr>
<td>Credits – 2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Course Learning Objectives: This course (18CSL37) will enable students to:

- Analog components and circuits including Operational Amplifier, Timer, etc.
- Combinational logic circuits.
- Flip - Flops and their operations
- Counters and registers using flip-flops.
- Synchronous and Asynchronous sequential circuits.
- A/D and D/A converters

Descriptions (if any):

- Simulation packages preferred: Multisim, Modelsim, PSpice or any other relevant.
- For Part A (Analog Electronic Circuits) students must trace the wave form on Tracing sheet / Graph sheet and label trace.
- Continuous evaluation by the faculty must be carried by including performance of a student in both hardware implementation and simulation (if any) for the given circuit.
- A batch not exceeding 4 must be formed for conducting the experiment. For simulation individual student must execute the program.

Laboratory Programs:

PART A (Analog Electronic Circuits)

1. Design an astable multivibrator circuit for three cases of duty cycle (50%, <50% and >50%) using NE 555 timer IC. Simulate the same for any one duty cycle.

2. Using ua 741 Opamp, design a 1 kHz Relaxation Oscillator with 50% duty cycle. And simulate the same.

3. Using ua 741 opamap, design a window comparator for any given UTP and LTP. And simulate the same.

PART B (Digital Electronic Circuits)

4. Design and implement Half adder, Full Adder, Half Subtractor, Full Subtractor using basic gates. And implement the same in HDL.

5. Given a 4-variable logic expression, simplify it using appropriate technique and realize the simplified logic expression using 8:1 multiplexer IC. And implement the same in HDL.

6. Realize a J-K Master / Slave Flip-Flop using NAND gates and verify its truth table. And implement the same in HDL.


8. Design and implement a mod-n (n<8) synchronous up counter using J-K Flip-Flop ICs and demonstrate its working.

9. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n (n<=9) and demonstrate on 7-segment display (using IC-7447)

Laboratory Outcomes: The student should be able to:

- Use appropriate design equations / methods to design the given circuit.
- Examine and verify the design of both analog and digital circuits using simulators.
- Make use of electronic components, ICs, instruments and tools for design and testing of circuits
for the given the appropriate inputs.

- Compile a laboratory journal which includes; aim, tool/instruments/software/components used, design equations used and designs, schematics, program listing, procedure followed, relevant theory, results as graphs and tables, interpreting and concluding the findings.

<table>
<thead>
<tr>
<th>Conduct of Practical Examination:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Experiment distribution</td>
</tr>
<tr>
<td>o For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.</td>
</tr>
<tr>
<td>o For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.</td>
</tr>
<tr>
<td>• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.</td>
</tr>
<tr>
<td>• Marks Distribution (<em>Courseed to change in accoradance with university regulations</em>)</td>
</tr>
<tr>
<td>a) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks</td>
</tr>
<tr>
<td>b) For laboratories having PART A and PART B</td>
</tr>
<tr>
<td>i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks</td>
</tr>
<tr>
<td>ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks</td>
</tr>
</tbody>
</table>
### course Objectives:

This course (18CSL38) will enable students to:

- Asymptotic performance of algorithms.
- Linear data structures and their applications such as stacks, queues and lists.
- Non-Linear data structures and their applications such as trees and graphs.
- Sorting and searching algorithms.

### Descriptions (if any):

- Implement all the programs in ‘C / C++’ Programming Language and Linux / Windows as OS.

### Programs List:

1. Design, Develop and Implement a menu driven Program in C for the following array operations.
   - a. Creating an array of N Integer Elements
   - b. Display of array Elements with Suitable Headings
   - c. Inserting an Element (ELEM) at a given valid Position (POS)
   - d. Deleting an Element at a given valid Position (POS)
   - e. Exit.
   
   Support the program with functions for each of the above operations.

2. Design, Develop and Implement a Program in C for the following operations on Strings.
   - a. Read a main String (STR), a Pattern String (PAT) and a Replace String (REP)
   - b. 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.
   
   Support the program with functions for each of the above operations. Don't use Built-in functions.

3. Design, Develop and Implement a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX).
   - a. Push an Element on to Stack
   - b. Pop an Element from Stack
   - c. Demonstrate how Stack can be used to check Palindrome
   - d. Demonstrate Overflow and Underflow situations on Stack
   - e. Display the status of Stack
   - f. Exit
   
   Support the program with appropriate functions for each of the above operations.

4. Design, Develop and Implement 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. Design, Develop and Implement a Program in C for the following Stack Applications.
   - a. Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^
   - b. Solving Tower of Hanoi problem with n disks.
6. Design, Develop and Implement a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX)
   a. Insert an Element on to Circular QUEUE
   b. Delete an Element from Circular QUEUE
   c. Demonstrate Overflow and Underflow situations on Circular QUEUE
   d. Display the status of Circular QUEUE
   e. Exit

Support the program with appropriate functions for each of the above operations

7. Design, Develop and Implement 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
   a. Create a SLL of N Students Data by using front insertion.
   b. Display the status of SLL and count the number of nodes in it
   c. Perform Insertion / Deletion at End of SLL
   d. Perform Insertion / Deletion at Front of SLL(Demonstration of stack)
   e. Exit

8. Design, Develop and Implement 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
   a. Create a DLL of N Employees Data by using end insertion.
   b. Display the status of DLL and count the number of nodes in it
   c. Perform Insertion and Deletion at End of DLL
   d. Perform Insertion and Deletion at Front of DLL
   e. Demonstrate how this DLL can be used as Double Ended Queue.
   f. Exit

9. Design, Develop and Implement a Program in C for the following operations on Singly Circular Linked List (SCLL) with header nodes
   a. Represent and Evaluate a Polynomial \( P(x,y,z) = 6x^2y^2z-4yz^5+3x^3yz+2xy^5z-2xyz^3 \)
   b. Find the sum of two polynomials \( POLY1(x,y,z) \) and \( POLY2(x,y,z) \) and store the result in \( POLYSUM(x,y,z) \)

Support the program with appropriate functions for each of the above operations

10. Design, Develop and Implement a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers
    a. Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2
    b. Traverse the BST in Inorder, Preorder and Post Order
    c. Search the BST for a given element (KEY) and report the appropriate message
    d. Exit

11. Design, Develop and Implement a Program in C for the following operations on Graph(G) of Cities
    a. Create a Graph of N cities using Adjacency Matrix.
    b. 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. Design and develop a Program in C that uses Hash function \( H: K \rightarrow L \) 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.

Laboratory Outcomes: The student should be able to:
- Analyze and Compare various linear and non-linear data structures
- Code, debug and demonstrate the working nature of different types of data structures and their applications
- Implement, analyze and evaluate the searching and sorting algorithms
- Choose the appropriate data structure for solving real world problems

<table>
<thead>
<tr>
<th>Conduct of Practical Examination:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Experiment distribution</td>
</tr>
<tr>
<td>o For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.</td>
</tr>
<tr>
<td>o For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.</td>
</tr>
<tr>
<td>• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.</td>
</tr>
<tr>
<td>• Marks Distribution (<em>Courseed to change in accordance with university regulations</em>)</td>
</tr>
<tr>
<td>c) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks</td>
</tr>
<tr>
<td>d) For laboratories having PART A and PART B</td>
</tr>
<tr>
<td>i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks</td>
</tr>
<tr>
<td>ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks</td>
</tr>
</tbody>
</table>
B. E. Common to all Programmes
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – II / III / IV
Aadalitha Kannada

Course Code: 18KAK28/39/49
Teaching Hours/Week (L:T:P): (0:2:0)
CIE Marks: 100
Credits: 01

**Course Details:**

- **Course Title:** Aadalitha Kannada
- **Course Code:** 18KAK28/39/49
- **Teaching Hours/Week:** (0:2:0)
- **Credits:** 01
- **CIE Marks:** 100

**Course Description:**

- **Course Overview:**
  - Aadalitha Kannada is a course that covers various aspects of Kannada literature and culture.
  - The course aims to provide students with a comprehensive understanding of Kannada language and its applications.
  - The course is designed to cater to students who are interested in pursuing higher studies in Kannada or those who wish to enhance their proficiency in the language.

**Learning Outcomes:**

- **Learning Outcome 1:** Students will be able to understand and interpret Kannada literature accurately.
- **Learning Outcome 2:** Students will develop a strong grasp of Kannada grammar and its applications in everyday communication.
- **Learning Outcome 3:** Students will be able to analyze and critique Kannada literary works effectively.

**Course Syllabus:**

- **Syllabus Overview:**
  - The syllabus covers various topics including Kannada literature, grammar, and cultural aspects.
  - The course includes both theoretical and practical components to ensure a comprehensive understanding of the subject.

**Assessment:**

- **Assessment Method:**
  - The assessment method includes internal evaluation, continuous internal evaluation, and final examination.

**Course Coordinator:**

- **Coordinator Name:** [Name]
- **Coordinator Contact:** [Contact Details]

**Inquiries:**

- For any questions or clarifications, please contact the course coordinator.
**Vyavaharika Kannada**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18KVK28/39/49</th>
<th>CIE Marks</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credits</td>
<td>01</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Course Learning Objectives:**
The course will enable the students to understand Kannada and communicate in Kannada language.

**Table of Contents:**
- Chapter - 1: Vyavaharika kannada – Parichaya (Introduction to Vyavaharika Kannada).
- Chapter - 2: Kannada Aksharamale haagu uchcharane ( Kannada Alpabets and Pronunciation).
- Chapter - 3: Sambhashanegaagi Kannada Padagalu (Kannada Vocabulary for Communication).
- Chapter - 4: Kannada Grammar in Conversations (Sambhashaneyalli Kannada Vyakarana).
- Chapter - 5: Activities in Kannada.

**Course Outcomes:**
At the end of the course, the student will be able to understand Kannada and communicate in Kannada language.
### B. E. Common to all Programmes

**Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**

**SEMESTER - III**

<table>
<thead>
<tr>
<th>CONSTITUTION OF INDIA, PROFESSIONAL ETHICS AND CYBER LAW (CPC)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Course Code</strong></td>
</tr>
<tr>
<td><strong>Teaching Hours/Week (L:T:P)</strong></td>
</tr>
<tr>
<td><strong>Credits</strong></td>
</tr>
</tbody>
</table>

**Course Learning Objectives:** To

- know the fundamental political codes, structure, procedures, powers, and duties of Indian government institutions, fundamental rights, directive principles, and the duties of citizens
- Understand engineering ethics and their responsibilities; identify their individual roles and ethical responsibilities towards society.
- Know about the cybercrimes and cyber laws for cyber safety measures.

### Module-1

**Introduction to Indian Constitution:**


### Module-2

**Union Executive and State Executive:**


### Module-3

**Elections, Amendments and Emergency Provisions:**


**Constitutional special provisions:**

Special Provisions for SC and ST, OBC, Women, Children and Backward Classes.

### Module-4

**Professional / Engineering Ethics:**


### Module-5

**Internet Laws, Cyber Crimes and Cyber Laws:**

Internet and Need for Cyber Laws, Modes of Regulation of Internet, Types of cyber terror capability, Net neutrality, Types of Cyber Crimes, India and cyber law, Cyber Crimes and the information Technology Act 2000, Internet Censorship, Cybercrimes and enforcement agencies.

**Course Outcomes:** On completion of this course, students will be able to,
CO 1: Have constitutional knowledge and legal literacy.
CO 2: Understand Engineering and Professional ethics and responsibilities of Engineers.
CO 3: Understand the cybercrimes and cyber laws for cyber safety measures.

**Question paper pattern for SEE and CIE:**
- The SEE question paper will be set for 100 marks and the marks scored by the students will proportionately be reduced to 60. The pattern of the question paper will be objective type (MCQ).
- For the award of 40 CIE marks, refer the University regulations 2018.

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Title of the Book</th>
<th>Name of the Author/s</th>
<th>Name of the Publisher</th>
<th>Edition and Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Textbook/s</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Constitution of India, Professional Ethics and Human Rights</td>
<td>Shubham Singles, Charles E. Haries, and et al</td>
<td>Cengage Learning India</td>
<td>2018</td>
</tr>
<tr>
<td>2</td>
<td>Cyber Security and Cyber Laws</td>
<td>Alfred Basta and et al</td>
<td>Cengage Learning India</td>
<td>2018</td>
</tr>
<tr>
<td>Reference Books</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Engineering Ethics</td>
<td>M. Govindarajan, S. Natarajan, V. S. Senthilkumar</td>
<td>Prentice –Hall,</td>
<td>2004</td>
</tr>
</tbody>
</table>
B. E. Common to all Programmes
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER - III

ADDITIONAL MATHEMATICS – I
(Mandatory Learning Course: Common to All Programmes)
(A Bridge course for Lateral Entry students under Diploma quota to BE/B. Tech. programmes)

Course Code: 18MATDIP31
CIE Marks: 40
Teaching Hours/Week (L:T:P): (2:2:0)
SEE Marks: 60
Credits: 0
Exam Hours: 03

Course Learning Objectives:
• To provide basic concepts of complex trigonometry, vector algebra, differential and integral calculus.
• To provide an insight into vector differentiation and first order ODE’s.

Module-1
Complex Trigonometry: Complex Numbers: Definitions and properties. Modulus and amplitude of a complex number, Argand’s diagram, De-Moivre’s theorem (without proof).

Module-2

Module-3

Module-4
Integral Calculus: Review of elementary integral calculus. Reduction formulae for sin^n x, cos^n x (with proof) and sin^m xcos^n x (without proof) and evaluation of these with standard limits-Examples. Double and triple integrals-Simple examples.

Module-5

Course Outcomes: At the end of the course the student will be able to:
• CO1: Apply concepts of complex numbers and vector algebra to analyze the problems arising in related area.
• CO2: Use derivatives and partial derivatives to calculate rate of change of multivariate functions.
• CO3: Analyze position, velocity and acceleration in two and three dimensions of vector valued functions.
• CO4: Learn techniques of integration including the evaluation of double and triple integrals.
• CO5: Identify and solve first order ordinary differential equations.

Question paper pattern:
• The question paper will have ten full questions carrying equal marks.
• Each full question will be for 20 marks.
• There will be two full questions (with a maximum of four sub-questions) from each module.
• Each full question will have sub-question covering all the topics under a module.
• The students will have to answer five full questions, selecting one full question from each module.

<p>| Sl No | Title of the Book | Name of the Author/s | Name of the Edition and Year |</p>
<table>
<thead>
<tr>
<th>Textbook</th>
<th>Publisher</th>
<th>Edition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Reference Books</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
B. E. COMMON TO ALL PROGRAMMES
Choice Based Credit System (CBCS) and Outcome Based Education (OBE)
SEMESTER - IV

COMPLEX ANALYSIS, PROBABILITY AND STATISTICAL METHODS
(Common to all programmes)
[As per Choice Based Credit System (CBCS) scheme]

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18MAT41</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teaching Hours/Week (L:T:P)</td>
<td>(2:2:0)</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Credits</td>
<td>03</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

Course Learning Objectives:
- To provide an insight into applications of complex variables, conformal mapping and special functions arising in potential theory, quantum mechanics, heat conduction and field theory.
- To develop probability distribution of discrete, continuous random variables and joint probability distribution occurring in digital signal processing, design engineering and microwave engineering.

Module-1


Construction of analytic functions: Milne-Thomson method-Problems.

Module-2

Conformal transformations: Introduction. Discussion of transformations:\( w = z^2, w = e^z, w = z + \frac{1}{z}, (z \neq 0) \). Bilinear transformations- Problems.

Complex integration: Line integral of a complex function-Cauchy’s theorem and Cauchy’s integral formula and problems.

Module-3

Probability Distributions: Review of basic probability theory. Random variables (discrete and continuous), probability mass/density functions. Binomial, Poisson, exponential and normal distributions- problems (No derivation for mean and standard deviation)-Illustrative examples.

Module-4


Curve Fitting: Curve fitting by the method of least squares- fitting the curves of the form-\( y = ax + b, y = ax^b \) and \( y = ax^2 + bx + c \).

Module-5

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

Sampling Theory: Introduction to sampling distributions, standard error, Type-I and Type-II errors. Test of hypothesis for means, student’s t-distribution, Chi-square distribution as a test of goodness of fit.

Course Outcomes:
- At the end of the course the student will be able to:
  - Use the concepts of analytic function and complex potentials to solve the problems arising in electromagnetic field theory.
  - Utilize conformal transformation and complex integral arising in aerofoil theory, fluid flow visualization and image processing.
  - Apply discrete and continuous probability distributions in analyzing the probability models arising in engineering field.
  - Make use of the correlation and regression analysis to fit a suitable mathematical model for the statistical data.
- Construct joint probability distributions and demonstrate the validity of testing the hypothesis.

**Question paper pattern:**
- The question paper will have ten full questions carrying equal marks.
- Each full question will be for 20 marks.
- There will be two full questions (with a maximum of four sub-questions) from each module.

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Title of the Book</th>
<th>Name of the Author/s</th>
<th>Name of the Publisher</th>
<th>Edition and Year</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Textbooks</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reference Books</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Advanced Engineering Mathematics</td>
<td>Chandrika Prasad and Reena Garg</td>
<td>Khanna Publishing,</td>
<td>2018</td>
</tr>
</tbody>
</table>

**Web links and Video Lectures:**
2. http://www.class-central.com/subject/math(MOOCs)
4. VTU EDUSAT PROGRAMME - 20
## DESIGN AND ANALYSIS OF ALGORITHMS

(Effective from the academic year 2018-2019)

**SEMESTER – IV**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS42</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:2:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>50</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 4**

**Course Learning Objectives:** This course (18CS42) will enable students to:
- Explain various computational problem solving techniques.
- Apply appropriate method to solve a given problem.
- Describe various methods of algorithm analysis.

### Module 1

**Introduction:** What is an Algorithm? (T2:1.1), Algorithm Specification (T2:1.2), Analysis Framework (T1:2.1), Performance Analysis: Space complexity, Time complexity (T2:1.3).

**Asymptotic Notations:** Big-Oh notation \( (O) \), Omega notation \( (\Omega) \), Theta notation \( (\Theta) \), and Little-oh notation \( (o) \). Mathematical analysis of Non-Recursive and recursive Algorithms with Examples (T1:2.2, 2.3, 2.4).

**Important Problem Types:** Sorting, Searching, String processing, Graph Problems, Combinatorial Problems.

**Fundamental Data Structures:** Stacks, Queues, Graphs, Trees, Sets and Dictionaries. (T1:1.3, 1.4).

**RBT:** L1, L2, L3

### Module 2

**Divide and Conquer:** General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum (T2:3.1, 3.3, 3.4), Merge sort, Quick sort (T1:4.1, 4.2), Strassen’s matrix multiplication (T2:3.8), Advantages and Disadvantages of divide and conquer. **Decrease and Conquer Approach:** Topological Sort. (T1:5.3).

**RBT:** L1, L2, L3

### Module 3

**Greedy Method:** General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines (T2:4.1, 4.3, 4.5).

**Minimum cost spanning trees:** Prim’s Algorithm, Kruskal’s Algorithm (T1:9.1, 9.2).

**Single source shortest paths:** Dijkstra’s Algorithm (T1:9.3).

**Optimal Tree problem:** Huffman Trees and Codes (T1:9.4).

**Transform and Conquer Approach:** Heaps and Heap Sort (T1:6.4).

**RBT:** L1, L2, L3

### Module 4

**Dynamic Programming:** General method with Examples, Multistage Graphs (T2:5.1, 5.2).

**Transitive Closure:** Warshall’s Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem ((T1:8.2, 8.3, 8.4), Bellman-Ford Algorithm (T2:5.4), Travelling Sales Person problem (T2:5.9), Reliability design (T2:5.8).

**RBT:** L1, L2, L3

### Module 5

**Backtracking:** General method (T2:7.1), N-Queens problem (T1:12.1), Sum of subsets problem (T1:12.1), Graph coloring (T2:7.4), Hamiltonian cycles (T2:7.5).

**Programme and Bound:** Assignment Problem, Travelling Sales Person problem (T1:12.2), Knapsack problem (T2:8.2, T1:12.2): LC Programme and Bound solution (T2:8.2), FIFO Programme and Bound solution (T2:8.2).

**NP-Complete and NP-Hard problems:** Basic concepts, non-deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes (T2:11.1).

**RBT:** L1, L2, L3

**Course Outcomes:** The student will be able to:
- Describe computational solution to well known problems like searching, sorting etc.
• Estimate the computational complexity of different algorithms.
• Devise an algorithm using appropriate design strategies for problem solving.

**Question Paper Pattern:**

• The question paper will have ten questions.
• Each full Question consisting of 20 marks
• There will be 2 full questions (with a maximum of four sub questions) from each module.
• Each full question will have sub questions covering all the topics under a module.
• The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**

# OPERATING SYSTEMS

**Course Code:** 18CS43  
**CIE Marks:** 40  
**Number of Contact Hours/Week:** 3:0:0  
**SEE Marks:** 60  
**Total Number of Contact Hours:** 40  
**Exam Hours:** 03  
**Module Hours:** 08

## Course Learning Objectives:
This course (18CS43) will enable students to:

- Introduce concepts and terminology used in OS
- Explain threading and multithreaded systems
- Illustrate process synchronization and concept of Deadlock
- Introduce Memory and Virtual memory management, File system and storage techniques

## Module 1

**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 generation; System boot. **Process Management** Process concept; Process scheduling; Operations on processes; Inter process communication

**Text book 1:** Chapter 1, 2.1, 2.3, 2.4, 2.5, 2.6, 2.8, 2.9, 2.10, 3.1, 3.2, 3.3, 3.4  
**RBT:** L1, L2, L3

## Module 2

**Multi-threaded Programming:** Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling. **Process Synchronization:** Synchronization: The critical section problem; Peterson’s solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

**Text book 1:** Chapter 4.1, 4.2, 4.3, 4.4, 5.1, 5.2, 5.3, 5.4, 5.5, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7  
**RBT:** L1, L2, L3

## Module 3

**Deadlocks:** Deadlocks; System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. **Memory Management:** Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation.

**Text book 1:** Chapter 7, 8.1 to 8.6  
**RBT:** L1, L2, L3

## Module 4

**Virtual Memory Management:** Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **File System, Implementation of File System:** File system: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection: Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management.

**Text book 1:** Chapter 91. To 9.6, 10.1 to 10.5

**RBT:** L1, L2, L3
Module 5

**Secondary Storage Structures, Protection:** Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability- Based systems.

**Case Study: The Linux Operating System:** Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory Management; File systems, Input and output; Inter-process communication.

**Text book 1:** Chapter 12.1 to 12.6, 21.1 to 21.9

RBT: L1, L2, L3

<table>
<thead>
<tr>
<th>Course Outcomes: The student will be able to:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Demonstrate need for OS and different types of OS</td>
</tr>
<tr>
<td>• Apply suitable techniques for management of different resources</td>
</tr>
<tr>
<td>• Use processor, memory, storage and file system commands</td>
</tr>
<tr>
<td>• Realize the different concepts of OS in platform of usage through case studies</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question Paper Pattern:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The question paper will have ten questions.</td>
</tr>
<tr>
<td>• Each full Question consisting of 20 marks</td>
</tr>
<tr>
<td>• There will be 2 full questions (with a maximum of four sub questions) from each module.</td>
</tr>
<tr>
<td>• Each full question will have sub questions covering all the topics under a module.</td>
</tr>
<tr>
<td>• The students will have to answer 5 full questions, selecting one full question from each module.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Textbooks:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reference Books:</th>
</tr>
</thead>
</table>
# Course Details

**Course Code**: 18CS44  
**CIE Marks**: 40  
**Number of Contact Hours/Week**: 3:0:0  
**SEE Marks**: 60  
**Total Number of Contact Hours**: 40  
**Exam Hours**: 03  
**Credits**: 3

**Course Learning Objectives**: This course (18CS44) will enable students to:
- Understand the fundamentals of ARM based systems, basic hardware components, selection methods and attributes of an embedded system.
- Program ARM controller using the various instructions.
- Identify the applicability of the embedded system.
- Comprehend the real time operating system used for the embedded system.

## Module 1

**Contact Hours**: 08

- ARM Processor Fundamentals: Registers, Current Program Status Register, Pipeline, Exceptions, Interrupts, and the Vector Table, Core Extensions.

**Text book 1**: Chapter 1 - 1.1 to 1.4, Chapter 2 - 2.1 to 2.5  
**RBT**: L1, L2

## Module 2

**Contact Hours**: 08

- Introduction to the ARM Instruction Set: Data Processing Instructions, Programme Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions, Loading Constants.
- ARM programming using Assembly language: Writing Assembly code, Profiling and cycle counting, instruction scheduling, Register Allocation, Conditional Execution, Looping Constructs.

**Text book 1**: Chapter 3:Sections 3.1 to 3.6 (Excluding 3.5.2), Chapter 6(Sections 6.1 to 6.6)  
**RBT**: L1, L2

## Module 3

**Contact Hours**: 08

- Embedded System Components: Embedded Vs General computing system, History of embedded systems, Classification of Embedded systems, Major applications areas of embedded systems, purpose of embedded systems.
- Core of an Embedded System including all types of processor/controller, Memory, Sensors, Actuators, LED, 7 segment LED display, stepper motor, Keyboard, Push button switch, Communication Interface (onboard and external types), Embedded firmware, Other system components.

**Text book 2**: Chapter 1(Sections 1.2 to 1.6), Chapter 2(Sections 2.1 to 2.6)  
**RBT**: L1, L2

## Module 4

**Contact Hours**: 08


**Text book 2**: Chapter 3, Chapter-4, Chapter-7 (Sections 7.1, 7.2 only), Chapter-9 (Sections 9.1, 9.2, 9.3.1, 9.3.2 only)  
**RBT**: L1, L2
| Module 5 |
|---------------------------------|------|
| **RTOS and IDE for Embedded System Design:** Operating System basics, Types of operating systems, Task, process and threads (Only POSIX Threads with an example program), Thread preemption, Multiprocessing and Multitasking, Task Communication (without any program), Task synchronization issues – Racing and Deadlock, Concept of Binary and counting semaphores (Mutex example without any program), How to choose an RTOS, Integration and testing of Embedded hardware and firmware, Embedded system Development Environment – Block diagram (excluding Keil), Disassembler/decompiler, simulator, emulator and debugging techniques, target hardware debugging, boundary scan. **Text book 2: Chapter-10 (Sections 10.1, 10.2, 10.3, 10.4, 10.7, 10.8.1.1, 10.8.1.2, 10.8.2.2, 10.10 only), Chapter 12, Chapter-13 (block diagram before 13.1, 13.3, 13.4, 13.5, 13.6 only)**  |
| **RBT:** L1, L2 |

**Course Outcomes:** The student will be able to:

- Describe the architectural features and instructions of ARM microcontroller
- Apply the knowledge gained for Programming ARM for different applications.
- Interface external devices and I/O with ARM microcontroller.
- Interpret the basic hardware components and their selection method based on the characteristics and attributes of an embedded system.
- Develop the hardware/software co-design and firmware design approaches.
- Demonstrate the need of real time operating system for embedded system applications.

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**

1. Raghunandan..G.H, Microcontroller (ARM) and Embedded System, Cengage learning Publication, 2019
<table>
<thead>
<tr>
<th>Module 1 Contact Hours</th>
<th>Introduction to Object Oriented Concepts:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Text book 1: Ch 1: 1.1 to 1.9 Ch 2: 2.1 to 2.3</td>
</tr>
<tr>
<td></td>
<td>RBT: L1, L2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Introduction to Java: Java’s magic: the Byte code; Java Development Kit (JDK); the Java Buzzwords, Object-oriented programming; Simple Java programs. Data types, variables and arrays, Operators, Control Statements.</td>
</tr>
<tr>
<td></td>
<td>Text book 1: Ch 2: 2.4 to 2.6 Ch 4: 4.1 to 4.2</td>
</tr>
<tr>
<td></td>
<td>Text book 2: Ch:1 Ch: 2 Ch:3 Ch:4 Ch:5</td>
</tr>
<tr>
<td></td>
<td>RBT: L1, L2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Classes, Inheritance, Exception Handling: Classes: Classes fundamentals; Declaring objects; Constructors, this keyword, garbage collection. Inheritance: inheritance basics, using super, creating multi level hierarchy, method overriding. Exception handling: Exception handling in Java.</td>
</tr>
<tr>
<td></td>
<td>Text book 2: Ch:6 Ch: 8 Ch:10</td>
</tr>
<tr>
<td></td>
<td>RBT: L1, L2, L3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Packages and Interfaces: Packages, Access Protection, Importing Packages, Interfaces. Multi Threaded Programming: Multi Threaded Programming: What are threads? How to make the classes threadable; Extending threads; Implementing runnable; Synchronization; Changing state of the thread; Bounded buffer problems, producer consumer problems.</td>
</tr>
<tr>
<td></td>
<td>Text book 2: CH: 9 Ch 11:</td>
</tr>
<tr>
<td></td>
<td>RBT: L1, L2, L3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 5</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Event Handling: Two event handling mechanisms; The delegation event model; Event classes; Sources of events; Event listener interfaces; Using the delegation event model; Adapter classes; Inner classes. Swings: Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application; Create a Swing Applet;</td>
</tr>
<tr>
<td>Jlabel and ImageIcon; JTextField; The Swing Buttons; JTabbedPane; JScrollPane; JList; JComboBox; JTable.</td>
<td>Text book 2: Ch 22: Ch 29: Ch 30  RBT: L1, L2, L3</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

**Course Outcomes:** The student will be able to:

- Explain the object-oriented concepts and JAVA.
- Develop computer programs to solve real world problems in Java.
- Develop simple GUI interfaces for a computer program to interact with users, and to understand the event-based GUI handling principles using swings.

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**


**Mandatory Note:** Every institute shall organize bridge course on C++, either in the vacation or in the beginning of even semester for a minimum period of ten days (2hrs/day). Maintain a copy of the report for verification during LIC visit.

Faculty can utilize open source tools to make teaching and learning more interactive.
<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Introduction:</strong> Data Communications, Networks, Network Types, Internet History, Standards and Administration, <strong>Networks Models:</strong> Protocol Layering, TCP/IP Protocol suite, The OSI model, <strong>Introduction to Physical Layer-1:</strong> Data and Signals, Digital Signals, Transmission Impairment, Data Rate limits, Performance.</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook1:</strong> Ch 1.1 to 1.5, 2.1 to 2.3, 3.1, 3.3 to 3.6</td>
<td></td>
</tr>
<tr>
<td><strong>RBT:</strong> L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Digital Transmission:</strong> Digital to digital conversion (Only Line coding: Polar, Bipolar and Manchester coding). <strong>Physical Layer-2:</strong> Analog to digital conversion (only PCM), Transmission Modes, <strong>Analog Transmission:</strong> Digital to analog conversion.</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook1:</strong> Ch 4.1 to 4.3, 5.1</td>
<td></td>
</tr>
<tr>
<td><strong>RBT:</strong> L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bandwidth Utilization:</strong> Multiplexing and Spread Spectrum, <strong>Switching:</strong> Introduction, Circuit Switched Networks and Packet switching, <strong>Error Detection and Correction:</strong> Introduction, Block coding, Cyclic codes, Checksum,</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook1:</strong> Ch 6.1, 6.2, 8.1 to 8.3, 10.1 to 10.4</td>
<td></td>
</tr>
<tr>
<td><strong>RBT:</strong> L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data link control:</strong> DLC services, Data link layer protocols, Point to Point protocol (Framing, Transition phases only). <strong>Media Access control:</strong> Random Access, Controlled Access and Channelization, <strong>Introduction to Data-Link Layer:</strong> Introduction, Link-Layer Addressing, ARP <strong>IPv4 Addressing and subnetting:</strong> Classful and CIDR addressing, DHCP, NAT</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook1:</strong> Ch 9.1, 9.2, 11.1, 11.2 11.4, 12.1 to 12.3, 18.4</td>
<td></td>
</tr>
<tr>
<td><strong>RBT:</strong> L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 5</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Wired LANs Ethernet:</strong> Ethernet Protocol, Standard Ethernet, Fast Ethernet, Gigabit</td>
<td>08</td>
</tr>
<tr>
<td>Course Outcomes: The student will be able to:</td>
<td></td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>• Explain the various components of data communication.</td>
<td></td>
</tr>
<tr>
<td>• Explain the fundamentals of digital communication and switching.</td>
<td></td>
</tr>
<tr>
<td>• Compare and contrast data link layer protocols.</td>
<td></td>
</tr>
<tr>
<td>• Summarize IEEE 802.xx standards</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question Paper Pattern:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The question paper will have ten questions.</td>
</tr>
<tr>
<td>• Each full question consisting of 20 marks</td>
</tr>
<tr>
<td>• There will be 2 full questions (with a maximum of four sub questions) from each module.</td>
</tr>
<tr>
<td>• Each full question will have sub questions covering all the topics under a module.</td>
</tr>
<tr>
<td>• The students will have to answer 5 full questions, selecting one full question from each module.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Textbooks:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reference Books:</th>
</tr>
</thead>
</table>
Course Learning Objectives: This course (18CSL47) will enable students to:

- Design and implement various algorithms in JAVA
- Employ various design strategies for problem solving.
- Measure and compare the performance of different algorithms.

Descriptions (if any):

- Design, develop, and implement the specified algorithms for the following problems using Java language under LINUX / Windows environment. Netbeans / Eclipse or IntelliJ Idea Community Edition IDE tool can be used for development and demonstration.
- Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.

Programs List:

1. 
   a. Create a Java class called Student with the following details as variables within it.
      (i) USN
      (ii) Name
      (iii) Programme
      (iv) Phone
      Write a Java program to create nStudent objects and print the USN, Name, Programme, and Phone of these objects with suitable headings.
   
   b. Write a Java program to implement the Stack using arrays. Write Push(), Pop(), and Display() methods to demonstrate its working.

2. 
   a. Design a superclass called Staff with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a Java program to read and display at least 3 staff objects of all three categories.
   
   b. Write a Java class called Customer to store their name and date_of_birth. The date_of_birth format should be dd/mm/yyyy. Write methods to read customer data as <name, dd/mm/yyyy> and display as <name, dd, mm, yyyy> using StringTokenizer class considering the delimiter character as “/”.

3. 
   a. Write a Java program to read two integers a and b. Compute \( \frac{a}{b} \) and print, when \( b \) is not zero. Raise an exception when \( b \) is equal to zero.
   
   b. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number.

4. 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 \( n \) on graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.
5. Sort a given set of \( n \) integer elements using \textbf{Merge Sort} method and compute its time complexity. Run the program for varied values of \( n \gg 5000 \), and record the time taken to sort. Plot a graph of the time taken versus non graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.

6. Implement in Java, the \textbf{0/1 Knapsack} problem using (a) Dynamic Programming method (b) Greedy method.

7. From a given vertex in a weighted connected graph, find shortest paths to other vertices using \textbf{Dijkstra's algorithm}. Write the program in Java.

8. Find Minimum Cost Spanning Tree of a given connected undirected graph using \textbf{Kruskal's algorithm}. Use Union-Find algorithms in your program.

9. Find Minimum Cost Spanning Tree of a given connected undirected graph using \textbf{Prim's algorithm}.

10. Write Java programs to (a) Implement All-Pairs Shortest Paths problem using \textbf{Floyd's algorithm}.
(b) Implement \textbf{Travelling Sales Person problem} using Dynamic programming.

11. Design and implement in Java to find a \textbf{subset} of a given set \( S = \{S_1, S_2, \ldots, S_n\} \) of \( n \) positive integers whose SUM is equal to a given positive integer \( d \). For example, if \( S = \{1, 2, 5, 6, 8\} \) and \( d = 9 \), there are two solutions \{1,2,6\}and \{1,8\}. Display a suitable message, if the given problem instance doesn't have a solution.

12. Design and implement in Java to find all \textbf{Hamiltonian Cycles} in a connected undirected Graph G of \( n \) vertices using backtracking principle.

**Laboratory Outcomes:** The student should be able to:

- Design algorithms using appropriate design techniques (brute-force, greedy, dynamic programming, etc.)
- Implement a variety of algorithms such as sorting, graph related, combinatorial, etc., in a high level language.
- Analyze and compare the performance of algorithms using language features.
- Apply and implement learned algorithm design techniques and data structures to solve real-world problems.

**Conduct of Practical Examination:**

- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (*Courseed to change in accordance with university regulations*)
  - For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
  - For laboratories having PART A and PART B
    - Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
    - Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks
# MICROCONTROLLER AND EMBEDDED SYSTEMS LABORATORY

(Effective from the academic year 2018 -2019)

## SEMESTER – IV

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CSL48</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIE Marks</td>
<td>40</td>
</tr>
<tr>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

## Credits – 2

### Course Learning Objectives:
This course (18CSL48) will enable students to:

- Develop and test Program using ARM7TDMI/LPC2148
- Conduct the experiments on an ARM7TDMI/LPC2148 evaluation board using evaluation version of Embedded 'C' & Keil Uvision-4 tool/compiler.

### Descriptions (if any):

### Programs List:

#### PART A
Conduct the following experiments by writing program using ARM7TDMI/LPC2148 using an evaluation board/simulator and the required software tool.

1. Write a program to multiply two 16 bit binary numbers.
2. Write a program to find the sum of first 10 integer numbers.
3. Write a program to find factorial of a number.
4. Write a program to add an array of 16 bit numbers and store the 32 bit result in internal RAM.
5. Write a program to find the square of a number (1 to 10) using look-up table.
6. Write a program to find the largest/smallest number in an array of 32 numbers.
7. Write a program to arrange a series of 32 bit numbers in ascending/descending order.
8. Write a program to count the number of ones and zeros in two consecutive memory locations.

#### PART –B
Conduct the following experiments on an ARM7TDMI/LPC2148 evaluation board using evaluation version of Embedded 'C' & Keil Uvision-4 tool/compiler.

9. Display “Hello World” message using Internal UART.
10. Interface and Control a DC Motor.
11. Interface a Stepper motor and rotate it in clockwise and anti-clockwise direction.
12. Determine Digital output for a given Analog input using Internal ADC of ARM controller.
13. Interface a DAC and generate Triangular and Square waveforms.
14. Interface a 4x4 keyboard and display the key code on an LCD.
15. Demonstrate the use of an external interrupt to toggle an LED On/Off.
16. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.

### Laboratory Outcomes:
The student should be able to:

- Develop and test Program using ARM7TDMI/LPC2148
- Conduct the following experiments on an ARM7TDMI/LPC2148 evaluation board using evaluation version of Embedded 'C' & Keil Uvision-4 tool/compiler.

### Conduct of Practical Examination:

- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (*Courseed to change in accoradance with university regulations*)
  - For laboratories having only one part – Procedure + Execution + Viva-Voce: $15+70+15 =$
100 Marks

h) For laboratories having PART A and PART B
   i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
   ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks
**B. E. Common to all Programmes**  
**Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  
**SEMESTER - IV**  
( Mandatory Learning Course: Common to All Programmes)  
( A Bridge course for Lateral Entry students under Diploma quota to BE/B. Tech. programmes)  

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18MATDIP41</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teaching Hours/Week (L:T:P)</td>
<td>(2:1:0)</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Credits</td>
<td>0</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**Course Learning Objectives:**
- To provide essential concepts of linear algebra, second & higher order differential equations along with methods to solve them.
- To provide an insight into elementary probability theory and numerical methods.

**Module-1**  

**Module-2**  
**Numerical Methods:** Finite differences. Interpolation/extrapolation using Newton’s forward and backward difference formulae (Statements only)-problems. Solution of polynomial and transcendental equations – Newton-Raphson and Regula-Falsi methods (only formulae)- Illustrative examples. Numerical integration: Simpson’s one third rule and Weddle’s rule (without proof) Problems.

**Module-3**  
**Higher order ODE’s:** Linear differential equations of second and higher order equations with constant coefficients. Homogeneous /non-homogeneous equations. Inverse differential operators. \[\text{Particular Integral restricted to } R(x) = e^{ax}, \sin ax /\cos ax \text{ for } f'(y) = R(x).\]

**Module-4**  
**Partial Differential Equations (PDE’s):** Formation of PDE’s by elimination of arbitrary constants and functions. Solution of non-homogeneous PDE by direct integration. Homogeneous PDEs involving derivative with respect to one independent variable only.

**Module-5**  
**Probability:** Introduction. Sample space and events. Axioms of probability. Addition & multiplication theorems. Conditional probability, Bayes’s theorem, problems.

**Course Outcomes:** At the end of the course the student will be able to:  
CO1: Solve systems of linear equations using matrix algebra.  
CO2: Apply the knowledge of numerical methods in modelling and solving engineering problems.  
CO3: Make use of analytical methods to solve higher order differential equations.  
CO4: Classify partial differential equations and solve them by exact methods.  
CO5: Apply elementary probability theory and solve related problems.

**Question paper pattern:**
- The question paper will have ten full questions carrying equal marks.  
- Each full question will be for 20 marks.  
- There will be two full questions (with a maximum of four sub-questions) from each module.  
- Each full question will have sub-question covering all the topics under a module.  
- The students will have to answer five full questions, selecting one full question from each module.
<table>
<thead>
<tr>
<th>Sl No</th>
<th>Title of the Book</th>
<th>Name of the Author/s</th>
<th>Name of the Publisher</th>
<th>Edition and Year</th>
</tr>
</thead>
</table>
## MANAGEMENT AND ENTREPRENEURSHIP FOR IT INDUSTRY
(Effective from the academic year 2018 -2019)

### SEMESTER – V

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS51</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

### CREDITS – 03

#### Course Learning Objectives:
This course (18CS51) will enable students to:
- Explain the principles of management, organization and entrepreneur.
- Discuss on planning, staffing, ERP and their importance
- Infer the importance of intellectual property rights and relate the institutional support

### Module – 1

**Introduction** - Meaning, nature and characteristics of management, scope and Functional areas of management, goals of management, levels of management, brief overview of evolution of management theories.
- Planning - Nature, importance, types of plans, steps in planning.
- Organizing - nature and purpose, types of Organization.
- Staffing - meaning, process of recruitment and selection.

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module – 2

**Directing and controlling** - meaning and nature of directing, leadership styles, motivation theories, Communication - Meaning and importance, Coordination - meaning and importance.
- Controlling - meaning, steps in controlling, methods of establishing control.

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module – 3

**Entrepreneur** – meaning of entrepreneur, characteristics of entrepreneurs, classification and types of entrepreneurs, various stages in entrepreneurial process, role of entrepreneurs in economic development, entrepreneurship in India and barriers to entrepreneurship.
- Identification of business opportunities, market feasibility study, technical feasibility study, financial feasibility study and social feasibility study.

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module – 4

**Preparation of project and ERP** - meaning of project, project identification, project selection, project report, need and significance of project report, contents, formulation, guidelines by planning commission for project report.

- **Enterprise Resource Planning:** Meaning and Importance.

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module – 5

**Micro and Small Enterprises:** Definition of micro and small enterprises, characteristics and advantages of micro and small enterprises, steps in establishing micro and small enterprises, Government of India industrial policy 2007 on micro and small enterprises, case study (Microsoft), Case study(Captain G R Gopinath), case study (N R Narayana Murthy & Infosys), Institutional support: MSME-DI, NSIC, SIDBI, KIADB, KSSIDC, TECSON, KSFC, DIC and District level single window agency, Introduction to IPR.

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>
**Course outcomes:** The students should be able to:

- Define management, organization, entrepreneur, planning, staffing, ERP and outline their importance in entrepreneurship
- Utilize the resources available effectively through ERP
- Make use of IPRs and institutional support in entrepreneurship

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**

2. Entrepreneurship Development - S S Khanka - S Chand & Co.
**COMPUTER NETWORKS AND SECURITY**  
(Effective from the academic year 2018 -2019)  
**SEMESTER – V**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS52</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:2:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>50</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS —4**

**Course Learning Objectives:** This course (18CS52) will enable students to:

- Demonstrate application layer protocols
- Discuss transport layer services and understand UDP and TCP protocols
- Explain routers, IP and Routing Algorithms in network layer
- Disseminate the Wireless and Mobile Networks covering IEEE 802.11 Standard
- Illustrate concepts of Multimedia Networking, Security and Network Management

**Module 1**

T1: Chap 2  
RBT: L1, L2, L3 | Contact Hours | 10 |

**Module 2**

T1: Chap 3  
RBT: L1, L2, L3 | Contact Hours | 10 |

**Module 3**

T1: Chap 4: 4.3-4.7  
RBT: L1, L2, L3 | Contact Hours | 10 |

**Module 4**

**Textbook2: Chapter 10**
RBT: L1, L2, L3

**Module 5**

**Textbook11: Chap 7**
RBT: L1, L2, L3

**Course Outcomes:** The student will be able to:
- Explain principles of application layer protocols
- Recognize transport layer services and infer UDP and TCP protocols
- Classify routers, IP and Routing Algorithms in network layer
- Understand the Wireless and Mobile Networks covering IEEE 802.11 Standard
- Describe Multimedia Networking and Network Management

**Question Paper Pattern:**
- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**

**Reference Books:**
2. Larry L Peterson and Brusce S Davie, Computer Networks, fifth edition, ELSEVIER
3. Andrew S Tanenbaum, Computer Networks, fifth edition, Pearson
## DATABASE MANAGEMENT SYSTEM

(Effective from the academic year 2018 -2019)

**SEMESTER – V**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS53</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:2:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>50</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –4**

**Course Learning Objectives:** This course (18CS53) will enable students to:
- Provide a strong foundation in database concepts, technology, and practice.
- Practice SQL programming through a variety of database problems.
- Demonstrate the use of concurrency and transactions in database
- Design and build database applications for real world problems.

### Module 1 Contact 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, database languages, and interfaces, The Database System environment. **Conceptual Data Modelling using Entities and Relationships:** Entity types, Entity sets, attributes, roles, and structural constraints, Weak entity types, ER diagrams, examples, Specialization and Generalization.

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

**RBT:** L1, L2, L3

10

### Module 2

**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.

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

**Textbook 1:** Ch4.1 to 4.5, 5.1 to 5.3, 6.1 to 6.5, 8.1; **Textbook 2:** 3.5

**RBT:** L1, L2, L3

10

### Module 3

**SQL : Advances Queries:** More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Views in SQL, Schema change statements in SQL. **Database Application Development:** Accessing databases from applications, An introduction to JDBC, JDBC classes and interfaces, SQLJ, Stored procedures, Case study: The internet Bookshop. **Internet Applications:** The three-Tier application architecture, The presentation layer, The Middle Tier

**Textbook 1:** Ch7.1 to 7.4; **Textbook 2:** 6.1 to 6.6, 7.5 to 7.7.

**RBT:** L1, L2, L3

10

### Module 4


10
Designs, Further discussion of Multivalued dependencies and 4NF, Other dependencies and Normal Forms

**Textbook 1:** Ch14.1 to 14.7, 15.1 to 15.6

**RBT:** L1, L2, L3

### Module 5

**Transaction Processing:** Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL.  
**Concurrency Control in Databases:** Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering, Multiversion Concurrency control techniques, Validation Concurrency control techniques, Granularity of Data items and Multiple Granularity Locking.  
**Introduction to Database Recovery Protocols:** Recovery Concepts, NO-UNDO/REDO recovery based on Deferred update, Recovery techniques based on immediate update, Shadow paging, Database backup and recovery from catastrophic failures

**Textbook 1:** 20.1 to 20.6, 21.1 to 21.7, 22.1 to 22.4, 22.7.  
**RBT:** L1, L2, L3

### Course Outcomes:

- Identify, analyze and define database objects, enforce integrity constraints on a database using RDBMS.  
- Use Structured Query Language (SQL) for database manipulation.  
- Design and build simple database systems  
- Develop application to interact with databases.

### Question Paper Pattern:

- The question paper will have ten questions.  
- Each full Question consisting of 20 marks  
- There will be 2 full questions (with a maximum of four sub questions) from each module.  
- Each full question will have sub questions covering all the topics under a module.  
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:


### Reference Books:

# AUTOMATA THEORY AND COMPUTABILITY

*(Effective from the academic year 2018-2019)*

**SEMIESTER – V**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS54</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 3**

**Course Learning Objectives:** This course (18CS54) will enable students to:

- Introduce core concepts in Automata and Theory of Computation
- Identify different Formal language Classes and their Relationships
- Design Grammars and Recognizers for different formal languages
- Prove or disprove theorems in automata theory using their properties
- Determine the decidability and intractability of Computational problems

## Module 1

**Why study the Theory of Computation, Languages and Strings:** Strings, Languages. A Language Hierarchy, Computation, **Finite State Machines (FSM):** Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs, From FSMs to Operational Systems, Simulators for FSMs, Minimizing FSMs, Canonical form of Regular languages, Finite State Transducers, Bidirectional Transducers.

**Textbook 1:** Ch 1, 2, 3, 4, 5.1 to 5.10  
**RBT:** L1, L2  
**Contact Hours** 08

## Module 2

**Regular Expressions (RE):** what is a RE?, Kleene’s theorem, Applications of REs, Manipulating and Simplifying REs. Regular Grammars: Definition, Regular Grammars and Regular languages. Regular Languages (RL) and Non-regular Languages: How many RLs, To show that a language is regular, Closure properties of RLs, to show some languages are not RLs.

**Textbook 1:** Ch 6, 7, 8: 6.1 to 6.4, 7.1, 7.2, 8.1 to 8.4  
**RBT:** L1, L2, L3  
**Contact Hours** 08

## Module 3

**Context-Free Grammars(CFG):** Introduction to Rewrite Systems and Grammars, CFGs and languages, designing CFGs, simplifying CFGs, proving that a Grammar is correct, Derivation and Parse trees, Ambiguity, Normal Forms. Pushdown Automata (PDA): Definition of non-deterministic PDA, Deterministic and Non-deterministic PDAs, Non-determinism and Halting, alternative equivalent definitions of a PDA, alternatives that are not equivalent to PDA.

**Textbook 1:** Ch 11, 12: 11.1 to 11.8, 12.1, 12.2, 12.4, 12.5, 12.6  
**RBT:** L1, L2, L3  
**Contact Hours** 08

## Module 4

**Algorithms and Decision Procedures for CFLs:** Decidable questions, Un-decidable questions. **Turing Machine:** Turing machine model, Representation, Language acceptability by TM, design of TM, Techniques for TM construction. Variants of Turing Machines (TM), The model of Linear Bounded automata.

**Textbook 1:** Ch 14: 14.1, 14.2, Textbook 2: Ch 9.1 to 9.8  
**RBT:** L1, L2, L3  
**Contact Hours** 08

## Module 5

**Decidability:** Definition of an algorithm, decidability, decidable languages, Undecidable languages, halting problem of TM, Post correspondence problem. Complexity: Growth rate of functions, the classes of P and NP, Quantum Computation: quantum computers, Church-
Turing thesis. **Applications:** G.1 Defining syntax of programming language, Appendix J: Security

**Textbook 2:** 10.1 to 10.7, 12.1, 12.2, 12.8, 12.8.1, 12.8.2

**Textbook 1:** Appendix: G.1(only), J.1 & J.2

**RBT:** L1, L2, L3

**Course Outcomes:** The student will be able to:

- Acquire fundamental understanding of the core concepts in automata theory and Theory of Computation
- Learn how to translate between different models of Computation (e.g., Deterministic and Non-deterministic and Software models).
- Design Grammars and Automata (recognizers) for different language classes and become knowledgeable about restricted models of Computation (Regular, Context Free) and their relative powers.
- Develop skills in formal reasoning and reduction of a problem to a formal model, with an emphasis on semantic precision and conciseness.
- Classify a problem with respect to different models of Computation.

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**

5. Basavaraj S. Anami, Karibasappa K G, Formal Languages and Automata theory, Wiley India, 2012

Faculty can utilize open source tools (like JFLAP) to make teaching and learning more interactive.
### APPLICATION DEVELOPMENT USING PYTHON

<table>
<thead>
<tr>
<th>Course Code</th>
<th>IA Marks</th>
<th>Exam Marks</th>
<th>CREDITS – 03</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS55</td>
<td>40</td>
<td>60</td>
<td></td>
</tr>
</tbody>
</table>

**Course Code 18CS55 IA Marks 40**

#### SEMESTER – V

**Number of Lecture Hours/Week**

<table>
<thead>
<tr>
<th>Total Number of Lecture Hours</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>03</td>
</tr>
</tbody>
</table>

**Number of Lecture Hours/Week**

**Total Number of Lecture Hours**

<table>
<thead>
<tr>
<th>CREDITS – 03</th>
</tr>
</thead>
<tbody>
<tr>
<td>03</td>
</tr>
</tbody>
</table>

**Course Learning Objectives:** This course (18CS55) will enable students to

- Learn the syntax and semantics of Python programming language.
- Illustrate the process of structuring the data using lists, tuples and dictionaries.
- Demonstrate the use of built-in functions to navigate the file system.
- Implement the Object Oriented Programming concepts in Python.
- Appraise the need for working with various documents like Excel, PDF, Word and Others.

#### Module – 1

**Teaching Hours**

**Python Basics**, Entering Expressions into the Interactive Shell, The Integer, Floating-Point, and String Data Types, String Concatenation and Replication, Storing Values in Variables, Your First Program, Dissecting Your Program, **Flow control**, Boolean Values, Comparison Operators, Boolean Operators, Elements of Flow Control, Program Execution, Flow Control Statements, Importing Modules, **Functions**, def Statements with Parameters, Return Values and return Statements, The None Value, Keyword Arguments and print(), Local and Global Scope, The global Statement, Exception Handling, A Short Program: Guess the Number

**Textbook 1: Chapters 1 – 3**

RBT: L1, L2

**Module – 2**

**Lists**, The List Data Type, Working with Lists, Augmented Assignment Operators, Methods, Example Program: Magic 8 Ball with a List, List-like Types: Strings and Tuples, References, **Dictionaries and Structuring Data**, The Dictionary Data Type, Pretty Printing, Using Data Structures to Model Real-World Things, **Manipulating Strings**, Working with Strings, Useful String Methods, Project: Password Locker, Project: Adding Bullets to Wiki Markup

**Textbook 1: Chapters 4 – 6**

RBT: L1, L2, L3

**Module – 3**

**Pattern Matching with Regular Expressions**, Finding Patterns of Text Without Regular Expressions, Finding Patterns of Text with Regular Expressions, More Pattern Matching with Regular Expressions, Greedy and Non-greedy Matching, The findall() Method, Character Classes, Making Your Own Character Classes, The Caret and Dollar Sign Characters, TheWildcard Character, Review of Regex Symbols, Case-Insensitive Matching, Substituting Strings with the sub() Method, Managing Complex Regexes, Combining re .IGNORECASE, re .DOTALL, and re .VERBOSE, Project: Phone Number and Email Address Extractor, **Reading and Writing Files**, Files and File Paths, The os, path Module, The File Reading/Writing Process, Saving Variables with the shelve Module, Saving Variables with the pprint.pformat() Function, Project: Generating Random Quiz Files, Project: Multiclipboard, **Organizing Files**, The shutil Module, Walking a Directory Tree, Compressing Files with the zipfile Module, Project: Renaming Files with American-Style Dates to European-Style Dates, Project: Backing Up a Folder into a ZIP File, **Debugging**, Raising Exceptions, Getting the Traceback as a String, Assertions, Logging, IDLE’s Debugger.

**Textbook 1: Chapters 7 – 10**
Module – 4

**Classes and objects**, Programmer-defined types, Attributes, Rectangles, Instances as return values, Objects are mutable, Copying, **Classes and functions**, Time, Pure functions, Modifiers, Prototyping versus planning, **Classes and methods**, Object-oriented features, Printing objects, Another example, A more complicated example, The `init` method, The `__str__` method, Operator overloading, Type-based dispatch, Polymorphism, Interface and implementation, **Inheritance**, Card objects, Class attributes, Comparing cards, Decks, Printing the deck, Add, remove, shuffle and sort, Inheritance, Class diagrams, Data encapsulation

**Textbook 2: Chapters 15 – 18**

Module – 5

**Web Scraping**, Project: MAPIT.PY with the webbrowser Module, Downloading Files from the Web with the requests Module, Saving Downloaded Files to the Hard Drive, HTML, Parsing HTML with the BeautifulSoup Module, Project: “I’m Feeling Lucky” Google Search, Project: Downloading All XKCD Comics, Controlling the Browser with the selenium Module, **Working with Excel Spreadsheets**, Excel Documents, Installing the openpyxl Module, Reading Excel Documents, Project: Reading Data from a Spreadsheet, Writing Excel Documents, Project: Updating a Spreadsheet, Setting the Font Style of Cells, Font Objects, Formulas, Adjusting Rows and Columns, Charts, **Working with PDF and Word Documents**, PDF Documents, Project: Combining Select Pages from Many PDFs, Word Documents, **Working with CSV files and JSON data**, The csv Module, Project: Removing the Header from CSV Files, JSON and APIs, The json Module, Project: Fetching Current Weather Data

**Textbook 1: Chapters 11 – 14**

Course Outcomes: After studying this course, students will be able to

- Demonstrate proficiency in handling of loops and creation of functions.
- Identify the methods to create and manipulate lists, tuples and dictionaries.
- Discover the commonly used operations involving regular expressions and file system.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Determine the need for scraping websites and working with CSV, JSON and other file formats.

**Question paper pattern:**

- The question paper will have ten questions.
- Each full question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**


**Reference Books:**

**UNIX PROGRAMMING**  
(Effective from the academic year 2018-2019)  
SEMESTER – V

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS56</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 3**

**Course Learning Objectives:** This course (18CS56) will enable students to

- Interpret the features of UNIX and basic commands.
- Demonstrate different UNIX files and permissions
- Implement shell programs.
- Explain UNIX process, IPC and signals.

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>
| **Introduction:** Unix Components/Architecture. Features of Unix. The UNIX Environment and UNIX Structure, Posix and Single Unix specification. General features of Unix commands/ command structure. Command arguments and options. Basic Unix commands such as echo, printf, ls, who, date, passwd, cal, Combining commands. Meaning of Internal and external commands. The type command: knowing the type of a command and locating it. The root login. Becoming the super user: su command.  
**RBT:** L1, L2 | 08 |

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>
| **File attributes and permissions:** The ls command with options. Changing file permissions: the relative and absolute permissions changing methods. Recursively changing file permissions. Directory permissions.  
**The shells interpretive cycle:** Wild cards. Removing the special meanings of wild cards. Three standard files and redirection. **Connecting commands:** Pipe. Basic and Extended regular expressions. The grep, egrep. Typical examples involving different regular expressions.  
**Shell programming:** Ordinary and environment variables. The .profile. Read and readonly commands. Command line arguments. exit and exit status of a command. Logical operators for conditional execution. The test command and its shortcut. The if, while, for and case control statements. The set and shift commands and handling positional parameters. The here (<<) document and trap command. Simple shell program examples.  
**RBT:** L1, L2 | 08 |

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>
| **UNIX File APIs:** General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs.  
**UNIX Processes and Process Control:**  
getrlimit, setrlimit Functions, UNIX Kernel Support for Processes.

**Process Control:** Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions

**RBT: L1, L2, L3**

**Module 4**

Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection.

**Overview of IPC Methods:** Pipes, popen, pclose Functions, Coprocesses, FIFOs, System V IPC, Message Queues, Semaphores.

**Shared Memory:** Client-Server Properties, Stream Pipes, Passing File Descriptors, An Open Server-Version 1, Client-Server Connection Functions.

**RBT: L1, L2, L3**

**Module 5**


**RBT: L1, L2, L3**

**Course Outcomes:** The student will be able to:

- Explain Unix Architecture, File system and use of Basic Commands
- Illustrate Shell Programming and to write Shell Scripts
- Categorize, compare and make use of Unix System Calls
- Build an application/service over a Unix system.

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**

1. Sumitabha Das., Unix Concepts and Applications., 4th Edition., Tata McGraw Hill ( Chapter 1,2,3,4,5,6,8,13,14)
3. Unix System Programming Using C++ - Terrence Chan, PHI, 1999. ( Chapter 7,8,9,10)

**Reference Books:**


Faculty can utilize open source tools to make teaching and learning more interactive.
Course Code 18CSL57  CIE Marks 40
Number of Contact Hours/Week 0:2:2  SEE Marks 60
Total Number of Lab Contact Hours 36  Exam Hours 03
Credits – 2

Course Learning Objectives: This course (18CSL57) will enable students to:

- Demonstrate operation of network and its management commands
- Simulate and demonstrate the performance of GSM and CDMA
- Implement data link layer and transport layer protocols.

Descriptions (if any):

- For the experiments below modify the topology and parameters set for the experiment and take multiple rounds of reading and analyze the results available in log files. Plot necessary graphs and conclude. Use NS2/NS3.
- Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.

Programs List:

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

PART B (Implement the following in Java)
7. Write a program for error detecting code using CRC-CCITT (16- bits).
8. Write a program to find the shortest path between vertices using bellman-ford algorithm.
9. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.
10. Write a program on datagram socket for client/server to display the messages on client side, typed at the server side.
11. Write a program for simple RSA algorithm to encrypt and decrypt the data.
12. Write a program for congestion control using leaky bucket algorithm.

Laboratory Outcomes: The student should be able to:

- Analyze and Compare various networking protocols.
- Demonstrate the working of different concepts of networking.
- Implement, analyze and evaluate networking protocols in NS2 / NS3 and JAVA programming language

Conduct of Practical Examination:

- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
- For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
  - Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
  - Marks Distribution (*Courseed to change in accoradance with university regulations*)
    i) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
    j) For laboratories having PART A and PART B
      i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
      ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks
Course Code: 18CSL58  CIE Marks: 40
Number of Contact Hours/Week: 0:2:2  SEE Marks: 60
Total Number of Lab Contact Hours: 36  Exam Hours: 03

Credits: 2

Course Learning Objectives: This course (18CSL58) will enable students to:

- Foundation knowledge in database concepts, technology and practice to groom students into well-informed database application developers.
- Strong practice in SQL programming through a variety of database problems.
- Develop database applications using front-end tools and back-end DBMS.

Descriptions (if any):

PART-A: SQL Programming (Max. Exam Mks. 50)
- Design, develop, and implement the specified queries for the following problems using Oracle, MySQL, MS SQL Server, or any other DBMS under LINUX/Windows environment.
- Create Schema and insert at least 5 records for each table. Add appropriate database constraints.

PART-B: Mini Project (Max. Exam Mks. 30)
- Use Java, C#, PHP, Python, or any other similar front-end tool. All applications must be demonstrated on desktop/laptop as a stand-alone or web based application (Mobile apps on Android/IOS are not permitted).

Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.

Programs List:

**PART A**

1. Consider the following schema for a Library Database:
   - BOOK(Book_id, Title, Publisher_Name, Pub_Year)
   - BOOK_AUTHORS(Book_id, Author_Name)
   - PUBLISHER(Name, Address, Phone)
   - BOOK_COPIES(Book_id, Programme_id, No-of_Copies)
   - BOOK_LENDING(Book_id, Programme_id, Card_No, Date_Out, Due_Date)
   - LIBRARY_PROGRAMME(Programme_id, Programme_Name, Address)

   Write SQL queries to
   1. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each Programme, etc.
   2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017.
   3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.
   4. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.
   5. Create a view of all books and its number of copies that are currently available in the Library.

2. Consider the following schema for Order Database:
   - SALESMAN(Salesman_id, Name, City, Commission)
   - CUSTOMER(Customer_id, Cust_Name, City, Grade, Salesman_id)
   - ORDERS(Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id)

   Write SQL queries to
1. Count the customers with grades above Bangalore’s average.
2. Find the name and numbers of all salesman who had more than one customer.
3. List all the salesman and indicate those who have and don’t have customers in their cities (Use UNION operation.)
4. Create a view that finds the salesman who has the customer with the highest order of a day.
5. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.

3. Consider the schema for Movie Database:
   - ACTOR(Act_id, Act_Name, Act_Gender)
   - DIRECTOR(Dir_id, Dir_Name, Dir_Phone)
   - MOVIES(Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id)
   - MOVIE_CAST(Act_id, Mov_id, Role)
   - RATING(Mov_id, Rev_Stars)

   Write SQL queries to
   1. List the titles of all movies directed by ‘Hitchcock’.
   2. Find the movie names where one or more actors acted in two or more movies.
   3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).
   4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.
   5. Update rating of all movies directed by ‘Steven Spielberg’ to 5.

4. Consider the schema for College Database:
   - STUDENT(USN, SName, Address, Phone, Gender)
   - SEMSEC(SSID, Sem, Sec)
   - CLASS(USN, SSID)
   - COURSE(Subcode, Title, Sem, Credits)
   - IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)

   Write SQL queries to
   1. List all the student details studying in fourth semester ‘C’ section.
   2. Compute the total number of male and female students in each semester and in each section.
   3. Create a view of Test1 marks of student USN ‘1BI15CS101’ in all Courses.
   4. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students.
   5. Categorize students based on the following criterion:
      - If FinalIA = 17 to 20 then CAT = ‘Outstanding’
      - If FinalIA = 12 to 16 then CAT = ‘Average’
      - If FinalIA< 12 then CAT = ‘Weak’
      - Give these details only for 8th semester A, B, and C section students.

5. Consider the schema for Company Database:
   - EMPLOYEE(SSN, Name, Address, Sex, Salary, SuperSSN, DNo)
   - DEPARTMENT(DNo, DName, MgrSSN, MgrStartDate)
   - DLOCATION(DNo, DLoc)
   - PROJECT(PNo, PName, PLocation, DNo)
   - WORKS_ON(SSN, PNo, Hours)

   Write SQL queries to
   1. Make a list of all project numbers for projects that involve an employee whose last name is ‘Scott’, either as a worker or as a manager of the department that controls the project.
2. Show the resulting salaries if every employee working on the ‘IoT’ project is given a 10 percent raise.
3. Find the sum of the salaries of all employees of the ‘Accounts’ department, as well as the maximum salary, the minimum salary, and the average salary in this department.
4. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator).
5. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.

PART B: Mini Project
- For any problem selected
- Make sure that the application should have five or more tables
- Indicative areas include; health care

Laboratory Outcomes: The student should be able to:
- Create, Update and query on the database.
- Demonstrate the working of different concepts of DBMS
- Implement, analyze and evaluate the project developed for an application.

Conduct of Practical Examination:
- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (*Courseed to change in accordance with university regulations*)
  k) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
  l) For laboratories having PART A and PART B
     i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
     ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks
B. E. COMMON TO ALL PROGRAMMES
Choice Based Credit System (CBCS) and Outcome Based Education (OBE)
SEMESTER – V
ENVIRONMENTAL STUDIES

Course Code: 18CIV59
CIE Marks: 40
Teaching Hours / Week (L:T:P): (1:0:0)
SEE Marks: 60
Credits: 01
Exam Hours: 02

Module - 1
Biodiversity: Types, Value; Hot-spots; Threats and Conservation of biodiversity, Forest Wealth, and Deforestation.

Module - 2
Natural Resource Management (Concept and case-studies): Disaster Management, Sustainable Mining, Cloud Seeding, and Carbon Trading.

Module - 3
Environmental Pollution (Sources, Impacts, Corrective and Preventive measures, Relevant Environmental Acts, Case-studies): Surface and Ground Water Pollution; Noise pollution; Soil Pollution and Air Pollution.
Waste Management & Public Health Aspects: Bio-medical Wastes; Solid waste; Hazardous wastes; E-wastes; Industrial and Municipal Sludge.

Module - 4
Global Environmental Concerns (Concept, policies and case-studies): Ground water depletion/recharging; Climate Change; Acid Rain; Ozone Depletion; Radon and Fluoride problem in drinking water; Resettlement and rehabilitation of people, Environmental Toxicology.

Module - 5
Field work: Visit to an Environmental Engineering Laboratory or Green Building or Water Treatment Plant or Waste water treatment Plant; ought to be Followed by understanding of process and its brief documentation.

Course Outcomes: At the end of the course, students will be able to:
- CO1: Understand the principles of ecology and environmental issues that apply to air, land, and water issues on a global scale,
- CO2: Develop critical thinking and/or observation skills, and apply them to the analysis of a problem or question related to the environment.
- CO3: Demonstrate ecology knowledge of a complex relationship between biotic and abiotic components.
- CO4: Apply their ecological knowledge to illustrate and graph a problem and describe the realities that managers face when dealing with complex issues.

Question paper pattern:
- The Question paper will have 100 objective questions.
- Each question will be for 01 marks
- Student will have to answer all the questions in an OMR Sheet.
- The Duration of Exam will be 2 hours.

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Title of the Book</th>
<th>Name of the Author/s</th>
<th>Name of the Publisher</th>
<th>Edition and Year</th>
</tr>
</thead>
</table>

Textbook/s
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Environmental Studies</td>
<td>S M Prakash</td>
<td>Pristine Publishing House, Mangalore</td>
<td>3rd Edition 2018</td>
</tr>
<tr>
<td>3</td>
<td>Environmental Studies – From Crisis to Cure</td>
<td>R Rajagopalan</td>
<td>Oxford Publisher</td>
<td>2005</td>
</tr>
</tbody>
</table>

**Reference Books**

<table>
<thead>
<tr>
<th>1</th>
<th>Principals of Environmental Science and Engineering</th>
<th>Raman Sivakumar</th>
<th>Cengage learning, Singapur.</th>
<th>2nd Edition, 2005</th>
</tr>
</thead>
</table>
# FILE STRUCTURES
(Effective from the academic year 2018-2019)

**SEMESTER – VI**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18IS61</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:2:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>50</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 4**

**Course Learning Objectives:** This course (18IS61) will enable students to:

- Explain the fundamentals of file structures and their management.
- Measure the performance of different file structures.
- Organize different file structures in the memory.
- Demonstrate hashing and indexing techniques.

## Module 1


**Module 2**

**Organization of Files for Performance, Indexing:** Data Compression, Reclaiming Space in files, Internal Sorting and Binary Searching, Keysorting; What is an Index? A Simple Index for Entry-Sequence File, Using Template Classes in C++ for Object I/O, Object-Oriented support for Indexed, Entry-Sequenced Files of Data Objects, Indexes that are too large to hold in Memory, Indexing to provide access by Multiple keys, Retrieval Using Combinations of Secondary Keys, Improving the Secondary Index structure: Inverted Lists, Selective indexes, Binding.

**Module 3**

**Consequential Processing and the Sorting of Large Files:** A Model for Implementing Consequential Processes, Application of the Model to a General Ledger Program, Extension of the Model to include Multiway Merging, A Second Look at Sorting in Memory, Merging as a Way of Sorting Large Files on Disk.

**Multi-Level Indexing and B-Trees:** The invention of B-Tree, Statement of the problem, Indexing with Binary Search Trees; Multi-Level Indexing, B-Trees, Example of Creating a B-Tree, An Object-Oriented Representation of B-Trees, B-Tree Methods; Nomenclature, Formal Definition of B-Tree Properties, Worst-case Search Depth, Deletion, Merging and Redistribution, Redistribution during insertion; B* Trees, Buffering of pages; Virtual B-Trees; Variable-length Records and keys.

**RBT: L1, L2, L3**
### Module 4


**RBT: L1, L2, L3**

### Module 5

**Hashing:** Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distribution, How much Extra Memory should be used?, Collision resolution by progressive overflow, Buckets, Making deletions, Other collision resolution techniques, Patterns of record access.

**Extendible Hashing:** How Extendible Hashing Works, Implementation, Deletion, Extendible Hashing Performance, Alternative Approaches.

**RBT: L1, L2, L3**

### Course Outcomes:

The student will be able to:

- Choose appropriate file structure for storage representation.
- Identify a suitable sorting technique to arrange the data.
- Select suitable indexing and hashing techniques for better performance to a given problem.

### Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:


### Reference Books:

| **SOFTWARE TESTING**  
(Effective from the academic year 2018-2019)  
SEMESTER – VI |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Course Code</strong></td>
<td>18IS62</td>
<td><strong>CIE Marks</strong></td>
</tr>
<tr>
<td><strong>Number of Contact Hours/Week</strong></td>
<td>3:2:0</td>
<td><strong>SEE Marks</strong></td>
</tr>
<tr>
<td><strong>Total Number of Contact Hours</strong></td>
<td>50</td>
<td><strong>Exam Hours</strong></td>
</tr>
</tbody>
</table>

**CREDITS –4**

**Course Learning Objectives:** This course (18IS62) will enable students to:

- Differentiate the various testing techniques
- Analyze the problem and derive suitable test cases.
- Apply suitable technique for designing of flow graph
- Explain the need for planning and monitoring a process

### Module 1

**Basics of Software Testing:** Basic definitions, Software Quality, Requirements, Behaviour and Correctness, Correctness versus Reliability, Testing and Debugging, Test cases, Insights from a Venn diagram, Identifying test cases, Test-generation Strategies, Test Metrics, Error and fault taxonomies, Levels of testing, Testing and Verification, Static Testing.

**Problem Statements:** Generalized pseudocode, the triangle problem, the NextDate function, the commission problem, the SATM (Simple Automatic Teller Machine) problem, the currency converter, Saturn windshield wiper

T1: Chapter 1, T3: Chapter 1, T1: Chapter 2.

RBT: L1, L2, L3

### Module 2

**Functional Testing:** Boundary value analysis, Robustness testing, Worst-case testing, Robust Worst testing for triangle problem, NextDate problem and commission problem, Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations, Decision tables, Test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations.

**Fault Based Testing:** Overview, Assumptions in fault based testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis.

T1: Chapter 5, 6 & 7, T2: Chapter 16

RBT: L1, L2, L3

### Module 3

**Structural Testing:** Overview, Statement testing, Programme testing, Condition testing, Path testing: DD paths, Test coverage metrics, Basis path testing, guidelines and observations, Data –Flow testing: Definition-Use testing, Slice-based testing, Guidelines and observations.

**Test Execution:** Overview of test execution, from test case specification to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay

T3: Section 6.2.1, T3: Section 6.2.4, T1: Chapter 9 & 10, T2: Chapter 17

RBT: L1, L2, L3

### Module 4

**Process Framework:** Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback, the quality process, Planning and monitoring, Quality goals, Dependability properties, Analysis Testing, Improving the process, Organizational factors.

**Planning and Monitoring the Process:** Quality and process, Test and analysis strategies and plans, Risk planning, monitoring the process, Improving the process, the quality team

**Documenting Analysis and Test:** Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.

<table>
<thead>
<tr>
<th><strong>Contact Hours</strong></th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Module 1</strong></td>
<td>10</td>
</tr>
<tr>
<td><strong>Module 2</strong></td>
<td>10</td>
</tr>
<tr>
<td><strong>Module 3</strong></td>
<td>10</td>
</tr>
<tr>
<td><strong>Module 4</strong></td>
<td>10</td>
</tr>
</tbody>
</table>
**Module 5**

**Integration and Component-Based Software Testing:** Overview, Integration testing strategies, Testing components and assemblies. System, Acceptance and Regression Testing: Overview, System testing, Acceptance testing, Usability, Regression testing, Regression test selection techniques, Test case prioritization and selective execution. **Levels of Testing,** **Integration Testing:** Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing, A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations.

**T2: Chapter 21 & 22, T1: Chapter 12 & 13**

**Course Outcomes:** The student will be able to:
- Derive test cases for any given problem
- Compare the different testing techniques
- Classify the problem into suitable testing model
- Apply the appropriate technique for the design of flow graph.
- Create appropriate document for the software artefact.

**Question Paper Pattern:**
- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**
1. Paul C. Jorgensen: Software Testing, A Craftsmen’s Approach, 3rd Edition, Auerbach Publications, 2008. (Listed topics only from Chapters 1, 2, 5, 6, 7, 9, 10, 12, 13)
3. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008. (Listed topics only from Section 1.2, 1.3, 1.4,1.5, 1.8,1.12,6, 2.1,6, 2.4)

**Reference Books:**
**WEB TECHNOLOGY AND ITS APPLICATIONS**  
(Effective from the academic year 2018 -2019)

**SEMESTER – VI**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS63</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:2:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>50</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –4**

**Course Learning Objectives:** This course (18CS63) will enable students to:

- Illustrate the Semantic Structure of HTML and CSS
- Compose forms and tables using HTML and CSS
- Design Client-Side programs using JavaScript and Server-Side programs using PHP
- Infer Object Oriented Programming capabilities of PHP
- Examine JavaScript frameworks such as jQuery and Backbone

**Module 1**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
</tr>
</tbody>
</table>


**Textbook 1:** Ch. 2, 3

**RBT:** L1, L2, L3

**Module 2**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
</tr>
</tbody>
</table>

HTML Tables and Forms, Introducing Tables, Styling Tables, Introducing Forms, Form Control Elements, Table and Form Accessibility, Microformats, Advanced CSS: Layout, Normal Flow, Positioning Elements, Floating Elements, Constructing Multicolumn Layouts, Approaches to CSS Layout, Responsive Design, CSS Frameworks.

**Textbook 1:** Ch. 4, 5

**RBT:** L1, L2, L3

**Module 3**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
</tr>
</tbody>
</table>


**Textbook 1:** Ch. 6, 8

**RBT:** L1, L2, L3

**Module 4**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
</tr>
</tbody>
</table>

PHP Arrays and Superglobals, Arrays, $_GET and $_POST Superglobal Arrays, $_SERVER Array, $_Files Array, Reading/Writing Files, PHP Classes and Objects, Object-Oriented Overview, Classes and Objects in PHP, Object Oriented Design, Error Handling and Validation, What are Errors and Exceptions?, PHP Error Reporting, PHP Error and Exception Handling

**Textbook 1:** Ch. 9, 10

**RBT:** L1, L2, L3

**Module 5**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
</tr>
</tbody>
</table>


**Textbook 1:** Ch. 11, 12

**RBT:** L1, L2, L3
Course Outcomes: The student will be able to:

- Adapt HTML and CSS syntax and semantics to build web pages.
- Construct and visually format tables and forms using HTML and CSS.
- Develop Client-Side Scripts using JavaScript and Server-Side Scripts using PHP to generate and display the contents dynamically.
- Appraise the principles of object oriented development using PHP.
- Inspect JavaScript frameworks like jQuery and Backbone which facilitates developer to focus on core features.

Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Textbooks:


Reference Books:


Mandatory Note:

Distribution of CIE Marks is a follows (Total 40 Marks):

- 20 Marks through IA Tests
- 20 Marks through practical assessment

Maintain a copy of the report for verification during LIC visit.

Possible list of practicals:

1. Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.
2. Write a JavaScript that calculates the squares and cubes of the numbers from 0 to 10 and outputs HTML text that displays the resulting values in an HTML table format.
3. Write a JavaScript code that displays text “TEXT-GROWING” with increasing font size in the interval of 100ms in RED COLOR, when the font size reaches 50pt it displays “TEXT-SHRINKING” in BLUE color. Then the font size decreases to 5pt.
4. Develop and demonstrate a HTML5 file that includes JavaScript script that uses functions for the following problems:
   a. Parameter: A string
   b. Output: The position in the string of the left-most vowel
c. Parameter: A number

d. Output: The number with its digits in the reverse order

5. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Programme, Year of Joining, and email id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.

6. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.

7. Write a PHP program to display a digital clock which displays the current time of the server.

8. Write the PHP programs to do the following:
   a. Implement simple calculator operations.
   b. Find the transpose of a matrix.
   c. Multiplication of two matrices.
   d. Addition of two matrices.

9. Write a PHP program named states.py that declares a variable states with value "Mississippi Alabama Texas Massachusetts Kansas". write a PHP program that does the following:
   a. Search for a word in variable states that ends in xas. Store this word in element 0 of a list named statesList.
   b. Search for a word in states that begins with k and ends in s. Perform a case-insensitive comparison. [Note: Passing re.I as a second parameter to method compile performs a case-insensitive comparison.] Store this word in element 1 of statesList.
   c. Search for a word in states that begins with M and ends in s. Store this word in element 2 of the list.
   d. Search for a word in states that ends in a. Store this word in element 3 of the list.

10. Write a PHP program to sort the student records which are stored in the database using selection sort.
## DATA MINING AND DATA WAREHOUSING
(Effective from the academic year 2018 -2019)

### SEMESTER – VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS641</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –3**

### Course Learning Objectives:
This course (18CS641) will enable students to:

- Define multi-dimensional data models.
- Explain rules related to association, classification and clustering analysis.
- Compare and contrast between different classification and clustering algorithms.

### Module 1

**Data Warehousing & modeling:** Basic Concepts: Data Warehousing: A multitier Architecture, Data warehouse models: Enterprise warehouse, Data mart and virtual warehouse, Extraction, Transformation and loading, Data Cube: A multidimensional data model, Stars, Snowflakes and Fact constellations: Schemas for multidimensional Data models, Dimensions: The role of concept Hierarchies, Measures: Their Categorization and computation, Typical OLAP Operations

**Textbook 2:** Ch.4.1,4.2
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 2

**Data warehouse implementation & Data mining:** Efficient Data Cube computation: An overview, Indexing OLAP Data: Bitmap index and join index, Efficient processing of OLAP Queries, OLAP server Architecture ROLAP versus MOLAP Versus HOLAP. : Introduction: What is data mining, Challenges, Data Mining Tasks, Data: Types of Data, Data Quality, Data Preprocessing, Measures of Similarity and Dissimilarity.

**Textbook 2:** Ch.4.4
**Textbook 1:** Ch.1.1,1.2,1.4, 2.1 to 2.4
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 3

**Association Analysis:** Association Analysis: Problem Definition, Frequent Item set Generation, Rule generation. Alternative Methods for Generating Frequent Item sets, FP-Growth Algorithm, Evaluation of Association Patterns.

**Textbook 1:** Ch 6.1 to 6.7 (Excluding 6.4)
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 4

**Classification:** Decision Trees Induction, Method for Comparing Classifiers, Rule Based Classifiers, Nearest Neighbor Classifiers, Bayesian Classifiers.

**Textbook 1:** Ch 4.3,4.6,5.1,5.2,5.3
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 5

**Clustering Analysis:** Overview, K-Means, Agglomerative Hierarchical Clustering, DBSCAN, Cluster Evaluation, Density-Based Clustering, Graph-Based Clustering, Scalable Clustering Algorithms.

**Textbook 1:** Ch 8.1 to 8.5, 9.3 to 9.5
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Course Outcomes:
The student will be able to:

- Identify data mining problems and implement the data warehouse.
- Write association rules for a given data pattern.
- Choose between classification and clustering solution.

**Question Paper Pattern:**
- The question paper will have ten questions.
- Each full question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**
2. Jiawei Han, Micheline Kamber, Jian Pei: Data Mining - Concepts and Techniques, 3rd Edition, Morgan Kaufmann Publisher, 2012.

**Reference Books:**
# OBJECT ORIENTED MODELING AND DESIGN

(Effective from the academic year 2018-2019)

SEMESTER – VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS642</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

CREDITS –3

**Course Learning Objectives:** This course (18CS642) will enable students to:

- Describe the concepts involved in Object-Oriented modelling and their benefits.
- Demonstrate concept of use-case model, sequence model and state chart model for a given problem.
- Explain the facets of the unified process approach to design and build a Software system.
- Translate the requirements into implementation for Object Oriented design.
- Choose an appropriate design pattern to facilitate development procedure.

### Module 1

Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived Data; Packages. State Modeling: Events, States, Transitions and Conditions, State Diagrams, State diagram behaviour.

**Text Book-1:** 4, 5

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 2

UseCase Modelling and Detailed Requirements: Overview; Detailed object-oriented Requirements definitions; System Processes-A use case/Scenario view; Identifying Input and outputs-The System sequence diagram; Identifying Object Behaviour-The state chart Diagram; Integrated Object-oriented Models.

**Text Book-2:** Chapter 6: Page 210 to 250

**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 3

Process Overview, System Conception and Domain Analysis: Process Overview: Development stages; Development life Cycle; System Conception: Devising a system concept; elaborating a concept; preparing a problem statement. Domain Analysis: Overview of analysis; Domain Class model: Domain state model; Domain interaction model; Iterating the analysis.

**Text Book-1:** Chapter 10, 11, and 12

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 4

Use case Realization :The Design Discipline within up iterations: Object Oriented Design- The Bridge between Requirements and Implementation; Design Classes and Design within Class Diagrams; Interaction Diagrams-Realizing Use Case and defining methods; Designing with Communication Diagrams; Updating the Design Class Diagram; Package Diagrams- Structuring the Major Components; Implementation Issues for Three-Layer Design.

**Text Book-2:** Chapter 8: page 292 to 346

**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 5

Design Patterns: Introduction; what is a design pattern?, Describing design patterns, the catalogue of design patterns, Organizing the catalogue, How design patterns solve design problems, how to select a design patterns, how to use a design pattern; Creational patterns: prototype and singleton (only); structural patterns adaptor and proxy (only).

**Text Book-3:** Ch-1: 1.1, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, Ch-3, Ch-4.
**Course Outcomes:** The student will be able to:

- Describe the concepts of object-oriented and basic class modelling.
- Draw class diagrams, sequence diagrams and interaction diagrams to solve problems.
- Choose and apply a befitting design pattern for the given problem.

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**

# CLOUD COMPUTING AND ITS APPLICATIONS
(Effective from the academic year 2018 -2019)

**SEMMESTER – VI**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS643</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

**Course Code: 18CS643 CIE Marks: 40 SEE Marks: 60 Total Number of Contact Hours: 40 Exam Hours: 03**

**Credits: 3**

**Course Learning Objectives:** This course (18CS643) will enable students to:
- Explain the fundamentals of cloud computing
- Illustrate the cloud application programming and aneka platform
- Contrast different cloud platforms used in industry

## Module 1

<table>
<thead>
<tr>
<th>Contact</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module 1</td>
<td>08</td>
</tr>
</tbody>
</table>


**Textbook:** Ch. 1, 3

**RBT:** L1, L2

## Module 2

<table>
<thead>
<tr>
<th>Contact</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module 2</td>
<td>08</td>
</tr>
</tbody>
</table>


**Textbook:** Ch. 4, 5

**RBT:** L1, L2

## Module 3

<table>
<thead>
<tr>
<th>Contact</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module 3</td>
<td>08</td>
</tr>
</tbody>
</table>


**Textbook 1: Ch. 6, 7**  
**RBT: L1, L2**

<table>
<thead>
<tr>
<th>Module 4</th>
</tr>
</thead>
</table>

**Textbook 1: Ch. 8**  
**RBT: L1, L2**

<table>
<thead>
<tr>
<th>Module 5</th>
</tr>
</thead>
</table>

**Textbook 1: Ch. 9,10**  
**RBT: L1, L2**

<table>
<thead>
<tr>
<th>Course Outcomes: The student will be able to:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Explain cloud computing, virtualization and classify services of cloud computing</td>
</tr>
<tr>
<td>• Illustrate architecture and programming in cloud</td>
</tr>
<tr>
<td>• Describe the platforms for development of cloud applications and List the application of cloud.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question Paper Pattern:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The question paper will have ten questions.</td>
</tr>
<tr>
<td>• Each full Question consisting of 20 marks</td>
</tr>
<tr>
<td>• There will be 2 full questions (with a maximum of four sub questions) from each module.</td>
</tr>
<tr>
<td>• Each full question will have sub questions covering all the topics under a module.</td>
</tr>
<tr>
<td>• The students will have to answer 5 full questions, selecting one full question from each module.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Textbooks:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reference Books:</th>
</tr>
</thead>
</table>
ADVANCED JAVA AND J2EE  
(Effective from the academic year 2018-2019)  
SEMESTER – VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS644</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

CREDITS – 3

**Course Learning Objectives:** This course (18CS644) will enable students to:
- Identify the need for advanced Java concepts like Enumerations and Collections
- Construct client-server applications using Java socket API
- Make use of JDBC to access database through Java Programs
- Adapt servlets to build server side programs
- Demonstrate the use of JavaBeans to develop component-based Java software

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enumerations, Autoboxing and Annotations (metadata): Enumerations, Enumeration fundamentals, the values() and valueOf() Methods, java enumerations are class types, enumerations Inherits Enum, example, type wrappers, Autoboxing, Autoboxing and Methods, Autoboxing/Unboxing occurs in Expressions, Autoboxing/Unboxing, Boolean and character values, Autoboxing/Unboxing helps prevent errors, A word of Warning. Annotations, Annotation basics, specifying retention policy, Obtaining Annotations at run time by use of reflection, Annotated element Interface, Using Default values, Marker Annotations, Single Member annotations, Built-In annotations.</td>
<td>08</td>
</tr>
</tbody>
</table>

**Textbook 1: Lesson 12**  
RBT: L1, L2, L3

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>The collections and Framework: Collections Overview, Recent Changes to Collections, The Collection Interfaces, The Collection Classes, Accessing a collection Via an Iterator, Storing User Defined Classes in Collections, The Random Access Interface, Working With Maps, Comparators, The Collection Algorithms, Why Generic Collections?, The legacy Classes and Interfaces, Parting Thoughts on Collections.</td>
<td>08</td>
</tr>
</tbody>
</table>

**Text Book 1: Ch.17**  
RBT: L1, L2, L3

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>String Handling: The String Constructors, String Length, Special String Operations, String Literals, String Concatenation, String Concatenation with Other Data Types, String Conversion and toString( ) Character Extraction, charAt( ), getChars( ), getBytes( ) toCharArray(), String Comparison, equals( ) and equalsIgnoreCase( ), regionMatches( ) startsWith( ) and endsWith( ), equals( ) Versus == , compareTo( ) Searching Strings, Modifying a String, substring( ), concat( ), replace( ), trim( ), Data Conversion Using valueOf( ), Changing the Case of Characters Within a String, Additional String Methods, StringBuffer, StringBuffer Constructors, length( ) and capacity( ), ensureCapacity( ), setLength( ), charAt( ) and setCharAt( ), getChars( ), append( ), insert( ), reverse( ), delete( ) and deleteCharAt( ), replace( ), substring( ), Additional StringBuffer Methods, StringBuilder</td>
<td>08</td>
</tr>
</tbody>
</table>

**Text Book 1: Ch 15**  
RBT: L1, L2, L3

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background: The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The Javax.servlet Package; Reading Servlet Parameter; The</td>
<td>08</td>
</tr>
</tbody>
</table>
### Course Outcomes:
The student will be able to:

- Interpret the need for advanced Java concepts like enumerations and collections in developing modular and efficient programs.
- Build client-server applications and TCP/IP socket programs.
- Illustrate database access and details for managing information using the JDBC API.
- Describe how servlets fit into Java-based web application architecture.
- Develop reusable software components using Java Beans.

### Question Paper Pattern:

- The question paper will have ten questions.
- Each full question consisting of 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:


### Reference Books:

# INFORMATION MANAGEMENT SYSTEM

(Effective from the academic year 2018 -2019)

**SEMESTER – VI**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18IS645</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

**Credits – 3**

**Course Learning Objectives:** This course (18IS645) will enable students to:

- Explain the Role of information management system in business
- Evaluate the role of the major types of information systems in a business environment and their relationship to each other

## Module 1


RBT: L1, L2, L3

## Module 2


RBT: L1, L2, L3

## Module 3


RBT: L1, L2, L3

## Module 4


RBT: L1, L2, L3

## Module 5

Decision support in business: Introduction, Decision support trends, Decision support systems (DSS), Management Information Systems, Online analytical processing, Using DSS, Executive information systems, Enterprise portals and decision support, Knowledge management systems, Business and Artificial Intelligence (AI), An overview of AI, Expert systems.

RBT: L1, L2, L3
**Course Outcomes:** The student will be able to:

- Describe the role of information technology and information systems in business
- Record the current issues of information technology and relate those issues to the firm
- Interpret how to use information technology to solve business problems

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**

## MOBILE APPLICATION DEVELOPMENT
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019)
SEMESTER – VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS651</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –3**

### Course Learning Objectives:
This course (18CS651) will enable students to:
- Learn to setup Android application development environment
- Illustrate user interfaces for interacting with apps and triggering actions
- Interpret tasks used in handling multiple activities
- Identify options to save persistent application data
- Appraise the role of security and performance in Android applications

<table>
<thead>
<tr>
<th>Module – 1</th>
<th>08</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get started, Build your first app, Activities, Testing, debugging and using support libraries</td>
<td>Textbook 1: Lesson 1,2,3</td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 2</th>
<th>08</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Interaction, Delightful user experience, Testing your UI</td>
<td>Textbook 1: Lesson 4,5,6</td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 3</th>
<th>08</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background Tasks, Triggering, scheduling and optimizing background tasks</td>
<td>Textbook 1: Lesson 7,8</td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 4</th>
<th>08</th>
</tr>
</thead>
<tbody>
<tr>
<td>All about data, Preferences and Settings, Storing data using SQLite, Sharing data with content providers, Loading data using Loaders</td>
<td>Textbook 1: Lesson 9,10,11,12</td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 5</th>
<th>08</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permissions, Performance and Security, Firebase and AdMob, Publish//</td>
<td>Textbook 1: Lesson 13,14,15</td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

### Course outcomes:
The students should be able to:
- Create, test and debug Android application by setting up Android development environment
- Implement adaptive, responsive user interfaces that work across a wide range of devices.
- Infer long running tasks and background work in Android applications
- Demonstrate methods in storing, sharing and retrieving data in Android applications
- Analyze performance of android applications and understand the role of permissions and security
- Describe the steps involved in publishing Android application to share with the world

### Question Paper Pattern:
- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>

**Reference Books:**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>
# INTRODUCTION TO DATA STRUCTURES AND ALGORITHM
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019)
SEMESTER – VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS652</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

## CREDITS –3

### Course Learning Objectives:
This course (18CS652) will enable students to:
- Identify different data structures in C programming language
- Appraise the use of data structures in problem solving
- Implement data structures using C programming language.

### Module 1
Introduction to C, constants, variables, data types, input output operations, operators and expressions, control statements, arrays, strings, built-in functions, user defined functions, structures, unions and pointers

**Text Book 1: Chapter 1 and 2**
**RBT: L1, L2**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 2
Algorithms, Asymptotic notations, Introduction to data structures, Types of data structures, Arrays.

**Text Book 1: Chapter 3 and 4**
**RBT: L1, L2**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 3
Linked lists, Stacks

**Text Book 1: Chapter 5 and 6**
**RBT: L1, L2**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 4
Queues, Trees

**Text Book 1: Chapter 7 and 8**
**RBT: L1, L2**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Module 5
Graphs, Sorting , (selection, insertion, bubble, quick) and searching (Linear, Binary, Hash)

**Text Book 1: Chapter 7 and 8**
**RBT: L1, L2**

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

### Course Outcomes:
The student will be able to :
- Identify different data structures in C programming language
- Appraise the use of data structures in problem solving
- Implement data structures using C programming language.

### Question Paper Pattern:
- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:

### Reference Books:
PROGRAMMING IN JAVA
(OPEN ELECTIVE)
(Effective from the academic year 2018-2019)
SEMESTER – VI

Course Code | 18CS653 | CIE Marks | 40
---|---|---|---
Number of Contact Hours/Week | 3:0:0 | SEE Marks | 60
Total Number of Contact Hours | 40 | Exam Hours | 03

CREDITS –3

**Course Learning Objectives:** This course (18CS653) will enable students to:

- Learn fundamental features of object oriented language and JAVA
- Set up Java JDK environment to create, debug and run simple Java programs.
- Learn object oriented concepts using programming examples.
- Study the concepts of importing of packages and exception handling mechanism.
- Discuss the String Handling examples with Object Oriented concepts

<table>
<thead>
<tr>
<th>Module – 1</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Text book 1:</strong> Ch 2, Ch 3</td>
<td></td>
</tr>
<tr>
<td><strong>RBT:</strong> L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 2</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Text book 1:</strong> Ch 4, Ch 5</td>
<td></td>
</tr>
<tr>
<td><strong>RBT:</strong> L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 3</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Text book 1:</strong> Ch 6, Ch 7.1-7.9, Ch 8.</td>
<td></td>
</tr>
<tr>
<td><strong>RBT:</strong> L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 4</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packages and Interfaces: Packages, Access Protection, Importing Packages, Interfaces, Exception Handling: 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, Using Exceptions.</td>
<td>08</td>
</tr>
</tbody>
</table>
### Module – 5

Enumerations, Type Wrappers, I/O, Applets, and Other Topics: I/O Basics, Reading Console Input, Writing Console Output, The PrintWriter Class, Reading and Writing Files, Applet Fundamentals, The transient and volatile Modifiers, Using instanceof, strictfp, Native Methods, Using assert, Static Import, Invoking Overloaded Constructors Through this(), String Handling: The String Constructors, String Length, Special String Operations, Character Extraction, String Comparison, Searching Strings, Modifying a String, Data Conversion Using valueOf(), Changing the Case of Characters Within a String, Additional String Methods, StringBuffer, StringBuilder.

### Course outcomes:

The students should be able to:

- Explain the object-oriented concepts and JAVA.
- Develop computer programs to solve real world problems in Java.

Develop simple GUI interfaces for a computer program to interact with users

### Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Text Books:


### Reference Books:

**INTRODUCTION TO OPERATING SYSTEM**  
*(OPEN ELECTIVE)*  
*(Effective from the academic year 2018 -2019)*  
**SEMESTER – VII**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS654</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –3**

**Course Learning Objectives:** This course (18CS654) will enable students to:

- Explain the fundamentals of operating system
- Comprehend multithreaded programming, process management, memory management and storage management.
- Familiar with various types of operating systems

<table>
<thead>
<tr>
<th>Module – 1</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction: What OS do, Computer system organization, architecture, structure, Operations, Process, memory and storage management, Protection and security, Distributed systems, Special purpose systems, computing environments.</td>
<td>08</td>
</tr>
<tr>
<td>System Structure: OS Services, User OSI, System calls, Types of system calls, System programs, OS design and implementation, OS structure, Virtual machines, OS generation, system boot</td>
<td></td>
</tr>
<tr>
<td>Textbook1: Chapter 1, 2</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 2</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Concept: Overview, Process scheduling, Operations on process, IPC, Examples in IPC, Communication in client-server systems.</td>
<td>08</td>
</tr>
<tr>
<td>Multithreaded Programming: Overview, Models, Libraries, Issues, OS Examples</td>
<td></td>
</tr>
<tr>
<td>Textbook1: Chapter 3,4</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 3</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Scheduling: Basic concept, Scheduling criteria, Algorithm, multiple processor scheduling, thread scheduling, OS Examples, Algorithm Evaluation.</td>
<td>08</td>
</tr>
<tr>
<td>Synchronization: Background, the critical section problem, Petersons solution, Synchronization hardware, Semaphores, Classic problems of synchronization, Monitors, Synchronization examples, Atomic transactions</td>
<td></td>
</tr>
<tr>
<td>Textbook1: Chapter 5, 6</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 4</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deadlocks: System model, Deadlock characterization, Method of handling deadlock, Deadlock prevention, Avoidance, Detection, Recovery from deadlock</td>
<td>08</td>
</tr>
<tr>
<td>Memory management strategies: Background, swapping, contiguous memory allocation,</td>
<td></td>
</tr>
</tbody>
</table>
Course outcomes: The students should be able to:

- Explain the fundamentals of operating system
- Comprehend process management, memory management and storage management.
- Familiar with various types of operating systems

Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. A. Silberschatz, P B Galvin, G Gagne, Operating systems, 7th edition, John Wiley and sons,.

Reference Books:

SOFTWARE TESTING LABORATORY
(Effective from the academic year 2018 -2019)
SEMESTER – VI

Course Code 18ISL66  CIE Marks 40
Number of Contact Hours/Week 0:2:2  SEE Marks 60
Total Number of Lab Contact Hours 36  Exam Hours 03

Credits – 2

Course Learning Objectives: This course (18ISL66) will enable students to:

- Analyse the requirements for the given problem statement
- Design and implement various solutions for the given problem
- Employ various design strategies for problem solving.
- Construct control flow graphs for the solution that is implemented
- Create appropriate document for the software artefact

Descriptions (if any):
Design, develop, and implement the specified algorithms for the following problems using any language of your choice under LINUX /Windows environment.

Programs List:

1. Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Assume that the upper limit for the size of any side is 10. Derive test cases for your program based on boundary-value analysis, execute the test cases and discuss the results.

2. Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of boundary value testing, derive different test cases, execute these test cases and discuss the test results.

3. Design, develop, code and run the program in any suitable language to implement the NextDate function. Analyze it from the perspective of boundary value testing, derive different test cases, execute these test cases and discuss the test results.

4. Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Assume that the upper limit for the size of any side is 10. Derive test cases for your program based on equivalence class partitioning, execute the test cases and discuss the results.

5. Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of equivalence class testing, derive different test cases, execute these test cases and discuss the test results.

6. Design, develop, code and run the program in any suitable language to implement the NextDate function. Analyze it from the perspective of equivalence class value testing, derive different test cases, execute these test cases and discuss the test results.

7. Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Derive test cases for your program based on decision-table approach, execute the test cases and discuss the results.

8. Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of decision table-based testing, derive different test cases, execute these test cases and discuss the test results.

9. Design, develop, code and run the program in any suitable language to solve the commission
10. Design, develop, code and run the program in any suitable language to implement the binary search algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

11. Design, develop, code and run the program in any suitable language to implement the quicksort algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

12. Design, develop, code and run the program in any suitable language to implement an absolute letter grading procedure, making suitable assumptions. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

**Laboratory Outcomes:** The student should be able to:
- List out the requirements for the given problem
- Design and implement the solution for given problem in any programming language (C, C++, JAVA)
- Derive test cases for any given problem
- Apply the appropriate technique for the design of flow graph.
- Create appropriate document for the software artefact.

**Conduct of Practical Examination:**
- All laboratory experiments, excluding the first, are to be included for practical examination.
- Experiment distribution
  - For questions having only one part: Students are allowed to pick one experiment from the lot and are given equal opportunity.
  - For questions having part A and B: Students are allowed to pick one experiment from part A and one experiment from part B and are given equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure part to be made zero.
- Marks Distribution (*Courseed to change in accoradance with university regulations*)
  - For questions having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
  - For questions having part A and B
    - i. Part A – Procedure + Execution + Viva = 4 + 21 + 5 = 30 Marks
    - ii. Part B – Procedure + Execution + Viva = 10 + 49 + 11 = 70 Marks
### Course Learning Objectives:

This course (18CISL67) will enable students to:

- Apply the concepts of Unix IPC to implement a given function.
- Measure the performance of different file structures.
- Write a program to manage operations on given file system.
- Demonstrate hashing and indexing techniques.

### Descriptions (if any):

### Programs List:

#### PART A

1. Write a program to read series of names, one per line, from standard input and write these names spelled in reverse order to the standard output using I/O redirection and pipes. Repeat the exercise using an input file specified by the user instead of the standard input and using an output file specified by the user instead of the standard output.

2. Write a program to read and write student objects with fixed-length records and the fields delimited by “|”. Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.

3. Write a program to read and write student objects with Variable - Length records using any suitable record structure. Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.

4. Write a program to write student objects with Variable - Length records using any suitable record structure and to read from this file a student record using RRN.

5. Write a program to implement simple index on primary key for a file of student objects. Implement add ( ), search ( ), delete ( ) using the index.

6. Write a program to implement index on secondary key, the name, for a file of student objects. Implement add ( ), search ( ), delete ( ) using the secondary index.

7. Write a program to read two lists of names and then match the names in the two lists using Consequential Match based on a single loop. Output the names common to both the lists.

8. Write a program to read k Lists of names and merge them using k-way merge algorithm with k = 8.

#### PART B MINI PROJECT

Students should develop mini project on the topics mentioned below or similar applications: Document processing, transaction management, indexing and hashing, buffer management, configuration management. Not limited to these.

**Laboratory Outcomes:** The student should be able to:

- Implement operations related to files.
- Apply the concepts of file system to produce the given application.
- Evaluate performance of various file systems on given parameters.

**Conduct of Practical Examination:**

- All laboratory experiments, excluding the first, are to be included for practical examination.
- Experiment distribution:
  - For questions having only one part: Students are allowed to pick one experiment from the lot and are given equal opportunity.
  - For questions having part A and B: Students are allowed to pick one experiment from part A and one experiment from part B and are given equal opportunity.
• Change of experiment is allowed only once and marks allotted for procedure part to be made zero.
• Marks Distribution (*Courseed to change in accordance with university regulations*)
  o) For questions having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
  p) For questions having part A and B
    i. Part A – Procedure + Execution + Viva = 4 + 21 + 5 = 30 Marks
    ii. Part B – Procedure + Execution + Viva = 10 + 49+ 11 = 70 Marks
## MOBILE APPLICATION DEVELOPMENT
(Effective from the academic year 2018 -2019)

**SEMESTER – VI**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CSMP68</th>
<th>IA Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>0:0:2</td>
<td>Exam Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>3 Hours/Week</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 02**

**Laboratory Objectives:** This laboratory (18CSMP68) will enable students to

- Learn and acquire the art of Android Programming.
- Configure Android studio to run the applications.
- Understand and implement Android's User interface functions.
- Create, modify and query on SQLite database.
- Inspect different methods of sharing data using services.

**Descriptions (if any):**

Installation procedure of the Android Studio/Java software must be demonstrated, carried out in groups.

Students should use the latest version of Android Studio/Java to execute these programs.

All of these diagrams are for representational purpose only. Students are expected to improvise on it.

**Programs List:**

### PART – A

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax and the website address is to be displayed. Insert a horizontal line between the job title and the phone number.</td>
</tr>
<tr>
<td>2</td>
<td>Develop an Android application using controls like Button, TextView, EditText for designing a calculator having basic functionality like Addition, Subtraction, Multiplication, and Division.</td>
</tr>
</tbody>
</table>
3. Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules:
   - Password should contain uppercase and lowercase letters.
   - Password should contain letters and numbers.
   - Password should contain special characters.
   - Minimum length of the password (the default value is 8).

On successful **SIGN UP** proceed to the next Login activity. Here the user should **SIGN IN** using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying “Successful Login” or else display a toast message saying “Login Failed”. The user is given only two attempts and after that display a toast message saying “Failed Login Attempts” and disable the **SIGN IN** button. Use Bundle to transfer information from one activity to another.

| 3 | Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules:  
|   | - Password should contain uppercase and lowercase letters.  
|   | - Password should contain letters and numbers.  
|   | - Password should contain special characters.  
|   | - Minimum length of the password (the default value is 8).  
|   |  
|   | On successful **SIGN UP** proceed to the next Login activity. Here the user should **SIGN IN** using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying “Successful Login” or else display a toast message saying “Login Failed”. The user is given only two attempts and after that display a toast message saying “Failed Login Attempts” and disable the **SIGN IN** button. Use Bundle to transfer information from one activity to another.  

4. Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds.

5. Write a program to create an activity with two buttons START and STOP. On pressing of the START button, the activity must start the counter by displaying the numbers from One and the counter must keep on counting until the STOP button is pressed. Display the counter value in a TextView control.
6. Create two files of XML and JSON type with values for City_Name, Latitude, Longitude, Temperature, and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective layouts side by side.

[Image: Parsing XML and JSON Data]

7. Develop a simple application with one EditText so that the user can write some text in it. Create a button called “Convert Text to Speech” that converts the user input text into voice.

[Image: Text to Speech Application]

8. Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts.

[Image: Call and Save Application]

---

PART - B

1. Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name.
2. Develop a content provider application with an activity called “Meeting Schedule” which takes Date, Time and Meeting Agenda as input from the user and store this information into the SQLite database. Create another application with an activity called “Meeting Info” having DatePicker control, which on the selection of a date should display the Meeting Agenda information for that particular date, else it should display a toast message saying “No Meeting on this Date”.

3. Create an application to receive an incoming SMS which is notified to the user. On clicking this SMS notification, the message content and the number should be displayed on the screen. Use appropriate emulator control to send the SMS message to your application.

4. Write a program to create an activity having a Text box, and also Save, Open and Create buttons. The user has to write some text in the Text box. On pressing the Create button the text should be saved as a text file in MkSDcard. On subsequent changes to the text, the Save button should be pressed to store the latest content to the same file. On pressing the Open button, it should display...
the contents from the previously stored files in the Text box. If the user tries to save the contents in the Textbox to a file without creating it, then a toast message has to be displayed saying “First Create a File”.

5 Create an application to demonstrate a basic media player that allows the user to Forward, Backward, Play and Pause an audio. Also, make use of the indicator in the seek bar to move the audio forward or backward as required.

6 Develop an application to demonstrate the use of Asynchronous tasks in android. The asynchronous task should implement the functionality of a simple moving banner. On pressing the Start Task button, the banner message should scroll from right to left. On pressing the Stop Task button, the banner message should stop. Let the banner message be “Demonstration of Asynchronous Task”.

7 Develop an application that makes use of the clipboard framework for copying and pasting of the text. The activity consists of two EditText controls and two Buttons to trigger the copy and paste functionality.
Create an AIDL service that calculates Car Loan EMI. The formula to calculate EMI is
\[ E = P \times \frac{(1+r)^n}{(1+r)^n-1} \]
where
- \( E \) = The EMI payable on the car loan amount
- \( P \) = The Car loan Principal Amount
- \( r \) = The interest rate value computed on a monthly basis
- \( n \) = The loan tenure in the form of months

The down payment amount has to be deducted from the principal amount paid towards buying the Car. Develop an application that makes use of this AIDL service to calculate the EMI. This application should have four EditText to read the Principal Amount, Down Payment, Interest Rate, Loan Term (in months) and a button named as “Calculate Monthly EMI”. On click of this button, the result should be shown in a TextView. Also, calculate the EMI by varying the Loan Term and Interest Rate values.

**Laboratory Outcomes:** After studying these laboratory programs, students will be able to
- Create, test and debug Android application by setting up Android development environment.
- Implement adaptive, responsive user interfaces that work across a wide range of devices.
- Infer long running tasks and background work in Android applications.
- Demonstrate methods in storing, sharing and retrieving data in Android applications.
- Infer the role of permissions and security for Android applications.

**Procedure to Conduct Practical Examination**
- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

- Marks Distribution (Course to change in accordance with university regulations)
  - For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
  - For laboratories having PART A and PART B
i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks

<table>
<thead>
<tr>
<th>Text Books:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reference Books:</th>
</tr>
</thead>
</table>
ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
(Effective from the academic year 2018 -2019)
SEMESTER – VII

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS71</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course Learning Objectives:</th>
</tr>
</thead>
<tbody>
<tr>
<td>This course (18CS71) will enable students to:</td>
</tr>
<tr>
<td>• Explain Artificial Intelligence and Machine Learning</td>
</tr>
<tr>
<td>• Illustrate AI and ML algorithm and their use in appropriate applications</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is artificial intelligence?, Problems, problem spaces and search, Heuristic search techniques</td>
<td>10</td>
</tr>
<tr>
<td>Textbook 1: Chapter 1, 2 and 3</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge representation issues, Predicate logic, Representation knowledge using rules.</td>
<td>10</td>
</tr>
<tr>
<td>Concept Learning: Concept learning task, Concept learning as search, Find-S algorithm, Candidate Elimination Algorithm, Inductive bias of Candidate Elimination Algorithm.</td>
<td></td>
</tr>
<tr>
<td>Textbook 1: Chapter 4, 5 and 6</td>
<td></td>
</tr>
<tr>
<td>Textbook2: Chapter 2 (2.1-2.5, 2.7)</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision Tree Learning: Introduction, Decision tree representation, Appropriate problems, ID3 algorithm.</td>
<td>10</td>
</tr>
<tr>
<td>Artificial Neural Network: Introduction, NN representation, Appropriate problems, Perceptrons, Backpropagation algorithm.</td>
<td></td>
</tr>
<tr>
<td>Textbook2: Chapter 3 (3.1-3.4), Chapter 4 (4.1-4.5)</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bayesian Learning: Introduction, Bayes theorem, Bayes theorem and concept learning, ML and LS error hypothesis, ML for predicting, MDL principle, Bates optimal classifier, Gibbs algorithm, Navie Bayes classifier, BBN, EM Algorithm</td>
<td>10</td>
</tr>
<tr>
<td>Textbook2: Chapter 6</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 5</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance-Base Learning: Introduction, k-Nearest Neighbour Learning, Locally weighted regression, Radial basis function, Case-Based reasoning. Reinforcement Learning: Introduction, The learning task, Q-Learning.</td>
<td>10</td>
</tr>
<tr>
<td>Textbook 1: Chapter 8 (8.1-8.5), Chapter 13 (13.1 – 13.3)</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

Course Outcomes: The student will be able to:
• Appaise the theory of Artificial intelligence and Machine Learning.
• Illustrate the working of AI and ML Algorithms.
• Demonstrate the applications of AI and ML.

Question Paper Pattern:

• The question paper will have ten questions.
• Each full question consisting of 20 marks
• There will be 2 full questions (with a maximum of four sub questions) from each module.
• Each full question will have sub questions covering all the topics under a module.
• The students will have to answer 5 full questions, selecting one full question from each module.

Textbooks:


Reference Books:

1. Saroj Kaushik, Artificial Intelligence, Cengage learning
5. Ethem Alpaydin, Introduction to machine learning, second edition, MIT press
# BIG DATA AND ANALYTICS

(Effective from the academic year 2018 -2019)

**SEMESTER – VII**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS72</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>4:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>50</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 4**

**Course Learning Objectives:** This course (18CS72) will enable students to:

- Understand fundamentals of Big Data analytics
- Explore the Hadoop framework and Hadoop Distributed File system
- Illustrate the concepts of NoSQL using MongoDB and Cassandra for Big Data
- Employ MapReduce programming model to process the big data
- Understand various machine learning algorithms for Big Data Analytics, Web Mining and Social Network Analysis.

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Big Data Analytics: Big Data, Scalability and Parallel Processing, Designing Data Architecture, Data Sources, Quality, Pre-Processing and Storing, Data Storage and Analysis, Big Data Analytics Applications and Case Studies.</td>
<td>10</td>
</tr>
<tr>
<td>Text book 1: Chapter 1: 1.2 -1.7</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Hadoop (T1): Introduction, Hadoop and its Ecosystem, Hadoop Distributed File System, MapReduce Framework and Programming Model, Hadoop Yarn, Hadoop Ecosystem Tools.</td>
<td>10</td>
</tr>
<tr>
<td>Hadoop Distributed File System Basics (T2): HDFS Design Features, Components, HDFS User Commands.</td>
<td></td>
</tr>
<tr>
<td>Essential Hadoop Tools (T2): Using Apache Pig, Hive, Sqoop, Flume, Oozie, HBase.</td>
<td></td>
</tr>
<tr>
<td>Text book 1: Chapter 2 :2.1-2.6</td>
<td></td>
</tr>
<tr>
<td>Text Book 2: Chapter 3</td>
<td></td>
</tr>
<tr>
<td>Text Book 2: Chapter 7 (except walk throughs)</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoSQL Big Data Management, MongoDB and Cassandra: Introduction, NoSQL Data Store, NoSQL Data Architecture Patterns, NoSQL to Manage Big Data, Shared-Nothing Architecture for Big Data Tasks, MongoDB, Databases, Cassandra Databases.</td>
<td>10</td>
</tr>
<tr>
<td>Text book 1: Chapter 3: 3.1-3.7</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>MapReduce, Hive and Pig: Introduction, MapReduce Map Tasks, Reduce Tasks and MapReduce Execution, Composing MapReduce for Calculations and Algorithms, Hive, HiveQL, Pig.</td>
<td>10</td>
</tr>
<tr>
<td>Text book 1: Chapter 4: 4.1-4.6</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 5</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine Learning Algorithms for Big Data Analytics: Introduction, Estimating the relationships, Outliers, Variances, Probability Distributions, and Correlations, Regression analysis, Finding Similar Items, Similarity of Sets and Collaborative Filtering, Frequent Itemsets and Association Rule Mining.</td>
<td>10</td>
</tr>
<tr>
<td>Text, Web Content, Link, and Social Network Analytics: Introduction, Text mining, Web</td>
<td></td>
</tr>
</tbody>
</table>
| Mining, Web Content and Web Usage Analytics, Page Rank, Structure of Web and analyzing a Web Graph, Social Network as Graphs and Social Network Analytics:  
| **Text book 1: Chapter 6: 6.1 to 6.5**  
| **Text book 1: Chapter 9: 9.1 to 9.5**  
| **Course Outcomes:** The student will be able to:  
| • Understand fundamentals of Big Data analytics.  
| • Investigate Hadoop framework and Hadoop Distributed File system.  
| • Illustrate the concepts of NoSQL using MongoDB and Cassandra for Big Data.  
| • Demonstrate the MapReduce programming model to process the big data along with Hadoop tools.  
| • Use Machine Learning algorithms for real world big data.  
| • Analyze web contents and Social Networks to provide analytics with relevant visualization tools.  
| **Question Paper Pattern:**  
| • The question paper will have ten questions.  
| • Each full Question consisting of 20 marks  
| • There will be 2 full questions (with a maximum of four sub questions) from each module.  
| • Each full question will have sub questions covering all the topics under a module.  
| • The students will have to answer 5 full questions, selecting one full question from each module.  
| **Textbooks:**  
| **Reference Books:**  
## Course Learning Objectives:
This course (18CS731) will enable students to:

- Learn how to add functionality to designs while minimizing complexity.
- What code qualities are required to maintain to keep code flexible?
- To understand the common design patterns.
- To explore the appropriate patterns for design problems.

### Module 1: Contact Hours

**Introduction**: what is a design pattern? describing design patterns, the catalog of design pattern, organizing the catalog, how design patterns solve design problems, how to select a design pattern, how to use a design pattern. A Notation for Describing Object-Oriented Systems

*Textbook 1: Chapter 1 and 2.7*

**Analysis a System**: overview of the analysis phase, stage 1: gathering the requirements functional requirements specification, defining conceptual classes and relationships, using the knowledge of the domain. Design and Implementation, discussions and further reading.

*Textbook 1: Chapter 6*

**Module 2**

**Design Pattern Catalog**: Structural patterns, Adapter, bridge, composite, decorator, facade, flyweight, proxy.

*Textbook 2: Chapter 4*

**Module 3**

**Behavioral Patterns**: Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Template Method

*Textbook 2: Chapter 5*

**Module 4**

**Interactive systems and the MVC architecture**: Introduction, The MVC architectural pattern, analyzing a simple drawing program, designing the system, designing of the subsystems, getting into implementation, implementing undo operation, drawing incomplete items, adding a new feature, pattern-based solutions.

*Textbook 1: Chapter 11*

**Module 5**

**Designing with Distributed Objects**: Client server system, java remote method invocation, implementing an object-oriented system on the web (discussions and further reading) a note on input and output, selection statements, loops arrays.

*Textbook 1: Chapter 12*

### Course Outcomes:
The student will be able to:
• Design and implement codes with higher performance and lower complexity
• Be aware of code qualities needed to keep code flexible
• Experience core design principles and be able to assess the quality of a design with respect to these principles.
• Capable of applying these principles in the design of object oriented systems.
• Demonstrate an understanding of a range of design patterns. Be capable of comprehending a design presented using this vocabulary.
• Be able to select and apply suitable patterns in specific contexts

**Question Paper Pattern:**

• The question paper will have ten questions.
• Each full Question consisting of 20 marks
• There will be 2 full questions (with a maximum of four sub questions) from each module.
• Each full question will have sub questions covering all the topics under a module.
• The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**

1. Brahma Dathan, Sarnath Rammath, Object-oriented analysis, design and implementation, Universities Press, 2013

**Reference Books:**

1. Frank Bachmann, Regine Meunier, Hans Rohnert “Pattern Oriented Software Architecture” – Volume 1, 1996.
# HIGH PERFORMANCE COMPUTING
(Effective from the academic year 2018-2019)

## SEMESTER – VII

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS732</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 3**

**Course Learning Objectives:** This course (18CS732) will enable students to:

- Introduce students the design, analysis, and implementation, of high performance computational science and engineering applications.
- Illustrate on advanced computer architectures, parallel algorithms, parallel languages, and performance-oriented computing.

### Module – 1


- **T1:** Ch: 1.1, 1.2, 2.1 – 2.7
- **RBT:** L1, L2

**Contact Hours: 08**

### Module – 2

**Principles of Parallel Algorithm Design:** Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models

**Basic Communication Operations:** One-to-All Broadcast and All-to-One Reduction, All-to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather, All-to-All Personalized Communication, Circular Shift, Improving the Speed of Some Communication Operations

- **T1:** Ch 3, 4
- **RBT:** L1, L2

**Contact Hours: 08**

### Module – 3

**Analytical Modeling of Parallel Programs:** Sources of Overhead in Parallel Programs, Performance Metrics for Parallel Systems, The Effect of Granularity on Performance, Scalability of Parallel Systems. Minimum Execution Time and Minimum Cost-Optimal Execution Time, Asymptotic Analysis of Parallel Programs Section 5.7. Other Scalability Metrics,


- **T1:** Ch 5, 6
- **RBT:** L1, L2, L3

**Contact Hours: 08**

### Module – 4

**Programming Shared Address Space Platforms:** Thread Basics, Why Threads?, The POSIX Thread API, Thread Basics: Creation and Termination, Synchronization Primitives in Pthreads, Controlling Thread and Synchronization Attributes, Thread Cancellation

**Contact Hours: 08**
| Composite Synchronization Constructs, Tips for Designing Asynchronous Programs, OpenMP: a Standard for Directive Based Parallel Programming |  
| Dense Matrix Algorithms: Matrix-Vector Multiplication, Matrix-Matrix Multiplication, Solving a System of Linear Equations |  

**T1: Ch 7, 8, 9**  
**RBT: L1, L2**  

**Module – 5**  

| 08  

**T1: Ch10, 11**  
**RBT: L1, L2**  

**Course outcomes:** The students should be able to:  
- Illustrate the key factors affecting performance of CSE applications  
- Illustrate mapping of applications to high-performance computing systems  
- Apply hardware/software co-design for achieving performance on real-world applications  

**Question paper pattern:**  
- The question paper will have ten questions.  
- There will be 2 questions from each module.  
- Each question will have questions covering all the topics under a module.  
- The students will have to answer 5 full questions, selecting one full question from each module.  

**Text Books:**  

**Reference Books:**  
## Course Code: 18CS733
### CIE Marks: 40
### SEE Marks: 60
### Total Number of Contact Hours: 40
### Exam Hours: 03
### Credits: 3

### Course Learning Objectives:
This course (18CS733) will enable students to:

- Describe computer architecture.
- Measure the performance of architectures in terms of right parameters.
- Summarize parallel architecture and the software used for them

### Module 1
**Contact Hours**
- **Theory of Parallelism:** Parallel Computer Models, The State of Computing, Multiprocessors and Multicomputer, Multivector and SIMD Computers, PRAM and VLSI Models, Program and Network Properties, Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanisms, System Interconnect Architectures, Principles of Scalable Performance, Performance Metrics and Measures, Parallel Processing Applications, Speedup Performance Laws. For all Algorithm or mechanism any one example is sufficient.
  - **Chapter 1 (1.1 to 1.4), Chapter 2 (2.1 to 2.4)**
  - RBT: L1, L2

### Module 2
**Contact Hours**
- **Hardware Technologies 1:** Processors and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology, Virtual Memory Technology. For all Algorithms or mechanisms any one example is sufficient.
  - **Chapter 4 (4.1 to 4.4)**
  - RBT: L1, L2, L3

### Module 3
**Contact Hours**
- **Hardware Technologies 2:** Bus Systems, Cache Memory Organizations, Shared Memory Organizations, Sequential and Weak Consistency Models, Pipelining and Superscalar Techniques, Linear Pipeline Processors, Nonlinear Pipeline Processors. For all Algorithms or mechanisms any one example is sufficient.
  - **Chapter 5 (5.1 to 5.4), Chapter 6 (6.1 to 6.2)**
  - RBT: L1, L2, L3

### Module 4
**Contact Hours**
  - **Chapter 7 (7.1, 7.2 and 7.4), Chapter 8 (8.1 to 8.3)**
  - RBT: L1, L2, L3

### Module 5
Software for parallel programming: Parallel Models, Languages, and Compilers, Parallel Programming Models, Parallel Languages and Compilers, Dependence Analysis of Data Arrays. Instruction and System Level Parallelism, Instruction Level Parallelism, Computer Architecture, Contents, Basic Design Issues, Problem Definition, Model of a Typical Processor, Compiler-detected Instruction Level Parallelism, Operand Forwarding, Reorder Buffer, Register Renaming, Tomasulo’s Algorithm. For all Algorithms or mechanisms any one example is sufficient.

Chapter 10(10.1 to 10.3) Chapter 12(12.1 to 12.9)
RBT: L1, L2, L3

Course Outcomes: The student will be able to:
- Explain the concepts of parallel computing and hardware technologies
- Compare and contrast the parallel architectures
- Illustrate parallel programming concepts

Question Paper Pattern:
- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Textbooks:

Reference Books:
# USER INTERFACE DESIGN

(Effective from the academic year 2018-2019)

**SEMESTER – VII**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>Number of Contact Hours/Week</th>
<th>SEE Marks</th>
<th>Total Number of Contact Hours</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS734</td>
<td>40</td>
<td>3:0:0</td>
<td>60</td>
<td>40</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 3**

**Course Learning Objectives:** This course (18CS734) will enable students to:

- To study the concept of menus, windows, interfaces
- To study about business functions
- To study the characteristics and components of windows and the various controls for the windows.
- To study about various problems in windows design with color, text, graphics and
- To study the testing methods

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>The User Interface-Introduction, Overview, The importance of user interface – Defining the user interface, The importance of Good design, Characteristics of graphical and web user interfaces, Principles of user interface design</td>
<td>08</td>
</tr>
<tr>
<td>Textbook 1: Ch. 1,2 RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>The User Interface Design process- Obstacles, Usability, Human characteristics in Design, Human Interaction speeds, Business functions-Business definition and requirement analysis, Basic business functions, Design standards.</td>
<td>08</td>
</tr>
<tr>
<td>Textbook 1: Part-2 RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>System menus and navigation schemes- Structures of menus, Functions of menus, Contents of menus, Formatting of menus, Phrasing the menu, Selecting menu choices, Navigating menus, Kinds of graphical menus.</td>
<td>08</td>
</tr>
<tr>
<td>Textbook 1: Part-2 RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows - Characteristics, Components of window, Window presentation styles, Types of window, Window management, Organizing window functions, Window operations, Web systems, Characteristics of device based controls.</td>
<td>08</td>
</tr>
<tr>
<td>Textbook 1: Part-2 RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 5</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Screen based controls- Operable control, Text control, Selection control, Custom control, Presentation control, Windows Tests-prototypes, kinds of tests.</td>
<td>08</td>
</tr>
<tr>
<td>Textbook 1: Part-2 RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

**Course Outcomes:** The student will be able to:

- Design the User Interface, design, menu creation, windows creation and connection between menus and windows

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Textbooks:**


**Reference Books:**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS741</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

CREDITS –3

Course Learning Objectives: This course (18CS741) will enable students to:

- Define the fundamental concepts in image processing
- Evaluate techniques followed in image enhancements
- Illustrate image segmentation and compression algorithms

Module 1

**Introduction**
Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Sampling and Quantization, Representing Digital Images (Data structure), Some Basic Relationships Between Pixels- Neighbors and Connectivity of pixels in image, Examples of fields that uses digital image processing

**Textbook 1:** Ch.1.3 to 1.5, Ch. 2.4,2.5
**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

Module 2

**Image Enhancement In The Spatial Domain:** Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods.

**Textbook 1:** Ch.3
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

Module 3

**Image Enhancement In Frequency Domain:** Introduction, Fourier Transform, Discrete Fourier Transform (DFT), properties of DFT , Discrete Cosine Transform (DCT), Image filtering in frequency domain.

**Textbook 1:** Ch.4.1,4.2
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

Module 4

**Image Segmentation:** Introduction, Detection of isolated points, line detection, Edge detection, Edge linking, Region based segmentation- Region growing, split and merge technique, local processing, regional processing, Hough transform, Segmentation using Threshold.

**Textbook 1:** Ch.10.1 to 10.3
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

Module 5

**Image Compression:** Introduction, coding Redundancy , Inter-pixel redundancy, image compression model, Lossy and Lossless compression, Huffman Coding, Arithmetic Coding, LZW coding, Transform Coding, Sub-image size selection, blocking, DCT implementation using FFT, Run length coding.

**Textbook 1:** Ch. 8.1 to 8.5
**RBT:** L1, L2, L3

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

Course Outcomes: The student will be able to:

- Explain fundamentals of image processing
- Compare transformation algorithms
- Contrast enhancement, segmentation and compression techniques
<table>
<thead>
<tr>
<th><strong>Question Paper Pattern:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• The question paper will have ten questions.</td>
</tr>
<tr>
<td>• Each full Question consisting of 20 marks.</td>
</tr>
<tr>
<td>• There will be 2 full questions (with a maximum of four sub questions) from each module.</td>
</tr>
<tr>
<td>• Each full question will have sub questions covering all the topics under a module.</td>
</tr>
<tr>
<td>• The students will have to answer 5 full questions, selecting one full question from each module.</td>
</tr>
<tr>
<td><strong>Textbooks:</strong></td>
</tr>
<tr>
<td><strong>Reference Books:</strong></td>
</tr>
</tbody>
</table>
# NETWORK MANAGEMENT

(Effective from the academic year 2018-2019)

**SEMESTER – VII**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS742</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
<tr>
<td>CREDITS –3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Course Learning Objectives:** This course (18CS742) will enable students to:

- Illustrate the need for interoperable network management.
- Explain the concepts and architecture behind standards based network management.
- Differentiate the concepts and terminology associated with SNMP and TMN
- Describe network management as a typical distributed application

## Module 1

**Introduction:**

**Textbook 1:** Ch.1

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

## Module 2

**Basic Foundations:** Standards, Models, and Language: Network Management Standards, Network Management Model, Organization Model, Information Model – Management Information Trees, Managed Object Perspectives, Communication Model; ASN.1- Terminology, Symbols, and Conventions, Objects and Data Types, Object Names, An Example of ASN.1 from ISO 8824; Encoding Structure; Macros, Functional Model.

**Textbook 1:** Ch.3

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

## Module 3

**SNMPv1 Network Management:**

**Textbook 1:** Ch. 4,5, Ch.8

**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
</tr>
</tbody>
</table>

## Module 4

Broadband Access Networks, Broadband Access Technology; HFCT Technology: The
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Textbook 1: Ch. 13</strong></td>
<td></td>
</tr>
<tr>
<td><strong>RBT: L1, L2</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Module 5</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Textbook 1: Ch.11</strong></td>
<td></td>
</tr>
<tr>
<td><strong>RBT: L1, L2</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Course Outcomes:</strong> The student will be able to:**</td>
<td></td>
</tr>
<tr>
<td>• Analyze the issues and challenges pertaining to management of emerging network technologies such as wired/wireless networks and high-speed internets.</td>
<td></td>
</tr>
<tr>
<td>• Apply network management standards to manage practical networks</td>
<td></td>
</tr>
<tr>
<td>• Formulate possible approaches for managing OSI network model.</td>
<td></td>
</tr>
<tr>
<td>• Use on SNMP for managing the network</td>
<td></td>
</tr>
<tr>
<td>• Use RMON for monitoring the behavior of the network</td>
<td></td>
</tr>
<tr>
<td>• Identify the various components of network and formulate the scheme for the managing them</td>
<td></td>
</tr>
<tr>
<td><strong>Question Paper Pattern:</strong></td>
<td></td>
</tr>
<tr>
<td>• The question paper will have ten questions.</td>
<td></td>
</tr>
<tr>
<td>• Each full Question consisting of 20 marks</td>
<td></td>
</tr>
<tr>
<td>• There will be 2 full questions (with a maximum of four sub questions) from each module.</td>
<td></td>
</tr>
<tr>
<td>• Each full question will have sub questions covering all the topics under a module.</td>
<td></td>
</tr>
<tr>
<td>• The students will have to answer 5 full questions, selecting one full question from each module.</td>
<td></td>
</tr>
<tr>
<td><strong>Textbooks:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Reference Books:</strong></td>
<td></td>
</tr>
</tbody>
</table>
NATURAL LANGUAGE PROCESSING
(Effective from the academic year 2018 -2019)
SEMESTER – VII

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS743</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

**Course Learning Objectives:** This course (18CS743) will enable students to:

**Module – 1**
Overview and language modeling:
- Textbook 1: Ch. 1,2
- RBT: L1, L2, L3

**Module – 2**
Word level and syntactic analysis:
- Textbook 1: Ch. 3,4
- RBT: L1, L2, L3

**Module – 3**
Extracting Relations from Text: From Word Sequences to Dependency Paths:
- Mining Diagnostic Text Reports by Learning to Annotate Knowledge Roles:
  - Introduction, Domain Knowledge and Knowledge Roles, Frame Semantics and Semantic Role Labeling, Learning to Annotate Cases with Knowledge Roles and Evaluations.
- Textbook 2: Ch. 3,4,5
- RBT: L1, L2, L3

**Module – 4**
- Introduction, iSTART: Feedback Systems, iSTART: Evaluation of Feedback Systems,
- Textual Signatures: Identifying Text-Types Using Latent Semantic Analysis to Measure the Cohesion of Text Structures:
- Evolving Explanatory Novel Patterns for Semantically-Based Text Mining: Related Work, A Semantically Guided Model for Effective Text Mining.
- Textbook 2: Ch. 6,7,8,9
## Module – 5


**Textbook 1: Ch. 9,12**

### Course outcomes:
The students should be able to:

- Analyze the natural language text.
- Define the importance of natural language.
- Understand the concepts Text mining.
- Illustrate information retrieval techniques.

### Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Text Books:


### Reference Books:

CRYPTOGRAPHY
(Effective from the academic year 2018 -2019)
SEMESTER – VII

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS744</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

CREDITS –3

Course Learning Objectives: This course (18CS744) will enable students to:

- Define cryptography and its principles
- Explain Cryptography algorithms
- Illustrate Public and Private key cryptography
- Explain Key management, distribution and ceritification
- Explain authentication protocols
- Tell about IPSec

Module – 1

Classical Encryption Techniques Symmetric Cipher Model, Cryptography, Cryptanalysis and Brute-Force Attack, Substitution Techniques, Caesar Cipher, Monoalphabetic Cipher, Playfair Cipher, Hill Cipher, Polyalphabetic Cipher, One Time Pad. Block Ciphers and the data encryption standard: Traditional block Cipher structure, stream Ciphers and block Ciphers, Motivation for the feistel Cipher structure, the feistel Cipher, The data encryption standard, DES encryption, DES decryption, A DES example, results, the avalanche effect, the strength of DES, the use of 56-Bit Keys, the nature of the DES algorithm, timing attacks, Block cipher design principles, number of rounds, design of function F, key schedule algorithm

Textbook 1: Ch. 2.1,2.2, Ch. 3
RBT: L1, L2

Module – 2

Public-Key Cryptography and RSA: Principles of public-key cryptosystems. Public-key cryptosystems. Applications for public-key cryptosystems, requirements for public-key cryptosystems. public-key cryptanalysis. The RSA algorithm, description of the algorithm, computational aspects, the security of RSA.

Other Public-Key Cryptosystems: Diffie-hellman key exchange, The algorithm, key exchange protocols, man in the middle attack,Elgamal Cryptographic systems

Textbook 1: Ch. 9, Ch. 10.1,10.2
RBT: L1, L2

Module – 3

Elliptic curve arithmetic, abelian groups, elliptic curves over real numbers, elliptic curves over Zp, elliptic curves over GF(2m), Elliptic curve cryptography, Analog of Diffie-hellman key exchange, Elliptic curve encryption/ decryption, security of Elliptic curve cryptography, Pseudorandom number generation based on an asymmetric cipher, PRNG based on RSA.

Key Management and Distribution: Symmetric key distribution using Symmetric encryption, A key distribution scenario, Hierarchical key control, session key lifetime, a transparent key control scheme, Decentralized key control, controlling key usage, Symmetric key distribution using asymmetric encryption, simple secret key distribution, secret key distribution with confidentiality and authentication, A hybrid scheme, distribution of public keys, public announcement of public keys, publicly available directory,public key authority, public keys certificates.
| Textbook 1: Ch. 10.3-10.5, Ch.14.1 to 14.3 |  
| RBT: L1, L2 |  
| **Module – 4** |  
| X-509 certificates. Certificates, X-509 version 3, public key infrastructure. **User Authentication:** Remote user Authentication principles, Mutual Authentication, one way Authentication, remote user Authentication using Symmetric encryption, Mutual Authentication, one way Authentication, Kerberos, Motivation, Kerberos version 4, Kerberos version 5, Remote user Authentication using Asymmetric encryption, Mutual Authentication, one way Authentication. **Electronic Mail Security:** Pretty good privacy, notation, operational; description, S/MIME, RFC5322, Multipurpose internet mail extensions, S/MIME functionality, S/MIME messages, S/MIME certificate processing, enhanced security services, Domain keys identified mail, internet mail architecture, E-Mail threats, DKIM strategy, DKIM functional flow. |  
| Textbook 1: Ch. 14.4, Ch. 15.1 to 15.4, Ch.19 |  
| RBT: L1, L2 |  
| **Module – 5** |  
| **Transport and tunnel modes,** combining security associations, authentication plus confidentiality, basic combinations of security associations, internet key exchange, key determinations protocol, header and payload formats, cryptographic suits. |  
| Textbook 1: Ch. 20.1 to 20.3 |  
| RBT: L1, L2 |  
| **Course outcomes:** The students should be able to: |  
| • Define cryptography and its principles |  
| • Explain Cryptography algorithms |  
| • Illustrate Public and Private key cryptography |  
| • Explain Key management, distribution and ceritification |  
| • Explain authentication protocols |  
| • Tell about IPSec |  
| **Question paper pattern:** |  
| • The question paper will have ten questions. |  
| • There will be 2 questions from each module. |  
| • Each question will have questions covering all the topics under a module. |  
| • The students will have to answer 5 full questions, selecting one full question from each module. |  
| **Text Books:** |  
| **Reference Books:** |  
# ROBOTIC PROCESS AUTOMATION DESIGN & DEVELOPMENT

(Effective from the academic year 2018 -2019)

**SEMESTER – VII**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS745</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –3**

**Course Learning Objectives:** This course (18CS745) will enable students to:

- To understand Basic Programming concepts and the underlying logic/structure
- To Describe RPA, where it can be applied and how its implemented
- To Describe the different types of variables, Control Flow and data manipulation techniques
- To Understand Image, Text and Data Tables Automation
- To Describe automation to Email and various types of Exceptions and strategies to handle

## Module – 1

**Contact Hours**

| RBT: L1, L2, L3 |

## Module – 2

| RBT: L1, L2, L3 |

## Module – 3

| RBT: L1, L2, L3 |

## Module – 4

Module – 5

Email Automation - Email Automation - Incoming Email automation - Sending Email automation - Debugging and Exception Handling - Debugging Tools - Strategies for solving issues - Catching errors.

RBT: L1, L2, L3

Course outcomes:
The students should be able to:

- To understand Basic Programming concepts and the underlying logic/structure
- To Describe RPA, where it can be applied and how its implemented
- To Describe the different types of variables, Control Flow and data manipulation techniques
- To Understand Image, Text and Data Tables Automation
- To Describe automation to Email and various types of Exceptions and strategies to handle

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

Reference Books:
### Course Code: 18CS751

**CCE Marks:** 40  
**SEE Marks:** 60  
**Total Number of Contact Hours:** 40  
**Exam Hours:** 03  
**CREDITS:** 3

**Course Learning Objectives:** This course (18CS751) will enable students to:
- Interpret the data in the context of the business.
- Identify an appropriate method to analyze the data.
- Show analytical model of a system.

### Module – 1


**Finding Relationships among Variables:** Introduction, Relationships among Categorical Variables, Relationships among Categorical Variables and a Numerical Variable, Stacked and Unstacked Formats, Relationships among Numerical Variables, Scatterplots, Correlation and Covariance.

**Textbook 1:** Ch. 1, 2, 3

**RBT:** L1, L2, L3  
**08 Teaching Hours**

### Module – 2


**Textbook 1:** Ch. 4, 5

**RBT:** L1, L2, L3  
**08 Teaching Hours**

### Module – 3

**Decision Making under Uncertainty:** Introduction, Elements of Decision Analysis, Payoff.

**Textbook 1:** Ch. 6

**RBT:** L1, L2, L3  
**08 Teaching Hours**
<table>
<thead>
<tr>
<th>Course outcomes: The students should be able to:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Explain the importance of data and data analysis</td>
</tr>
<tr>
<td>• Interpret the probabilistic models for data</td>
</tr>
</tbody>
</table>
- Define hypothesis, uncertainty principle
- Evaluate regression analysis

**Question Paper Pattern:**

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**

1. S C Albright and W L Winston, Business analytics: data analysis and decision making, 5/e Cenage Learning

**Reference Books:**

# PYTHON APPLICATION PROGRAMMING

(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019)

**SEMESTER – VI**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>IA Marks</th>
<th>Exam Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS752</td>
<td>40</td>
<td>60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course Learning Objectives:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>This course (18CS752) will enable students to</td>
<td></td>
</tr>
<tr>
<td>• Learn Syntax and Semantics and create Functions in Python.</td>
<td></td>
</tr>
<tr>
<td>• Handle Strings and Files in Python.</td>
<td></td>
</tr>
<tr>
<td>• Understand Lists, Dictionaries and Regular expressions in Python.</td>
<td></td>
</tr>
<tr>
<td>• Implement Object Oriented Programming concepts in Python</td>
<td></td>
</tr>
<tr>
<td>• Build Web Services and introduction to Network and Database Programming in Python.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 1</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Why should you learn to write programs, Variables, expressions and statements, Conditional execution, Functions</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook 1: Chapters 1 – 4</strong></td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Iteration, Strings, Files</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook 1: Chapters 5– 7</strong></td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 3</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Lists, Dictionaries, Tuples, Regular Expressions</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook 1: Chapters 8 - 11</strong></td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 4</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Classes and objects, Classes and functions, Classes and methods</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook 2: Chapters 15 – 17</strong></td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module – 5</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Networked programs, Using Web Services, Using databases and SQL</td>
<td>08</td>
</tr>
<tr>
<td><strong>Textbook 1: Chapters 12– 13, 15</strong></td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2, L3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course Outcomes:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>After studying this course, students will be able to</td>
<td></td>
</tr>
<tr>
<td>• Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.</td>
<td></td>
</tr>
<tr>
<td>• Demonstrate proficiency in handling Strings and File Systems.</td>
<td></td>
</tr>
<tr>
<td>• Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.</td>
<td></td>
</tr>
<tr>
<td>• Interpret the concepts of Object-Oriented Programming as used in Python.</td>
<td></td>
</tr>
<tr>
<td>• Implement exemplary applications related to Network Programming, Web Services and Databases in Python.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question paper pattern:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The question paper will have ten questions.</td>
<td></td>
</tr>
<tr>
<td>Each full Question consisting of 20 marks</td>
<td></td>
</tr>
<tr>
<td>There will be 2 full questions (with a maximum of four sub questions) from each module.</td>
<td></td>
</tr>
</tbody>
</table>
• Each full question will have sub questions covering all the topics under a module.
• The students will have to answer 5 full questions, selecting one full question from each module.

<table>
<thead>
<tr>
<th>Text Books:</th>
</tr>
</thead>
<tbody>
<tr>
<td>pythonlearn.pdf)</td>
</tr>
<tr>
<td>files from the above links)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reference Books:</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Roberto Tamassia, Michael H Goldwasser, Michael T Goodrich, “Data Structures and Algorithms in</td>
</tr>
</tbody>
</table>
**INTRODUCTION TO ARTIFICIAL INTELLIGENCE**
*(OPEN ELECTIVE)*
*(Effective from the academic year 2018 -2019)*
**SEMESTER – VII**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS753</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –3**

**Course Learning Objectives:** This course (18CS753) will enable students to:
- Identify the problems where AI is required and the different methods available.
- Compare and contrast different AI techniques available.
- Define and explain learning algorithms.

<table>
<thead>
<tr>
<th>Module – 1</th>
<th>Teaching Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is artificial intelligence?, Problems, Problem Spaces and search</td>
<td>08</td>
</tr>
<tr>
<td>TextBook1: Ch 1, 2</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

| Module – 2 | |
|------------||
| Knowledge Representation Issues, Using Predicate Logic, Representing knowledge using Rules. | 08 |
| TextBook1: Ch 4, 5 and 6. | |
| RBT: L1, L2 | |

| Module – 3 | |
|------------||
| Symbolic Reasoning under Uncertainty, Statistical reasoning | 08 |
| TextBook1: Ch 7, 8 | |
| RBT: L1, L2 | |

| Module – 4 | |
|------------||
| Game Playing, Natural Language Processing | 08 |
| TextBook1: Ch 12 and 15 | |
| RBT: L1, L2 | |

| Module – 5 | |
|------------||
| Learning, Expert Systems. | 08 |
| TextBook1: Ch 17 and 20 | |
| RBT: L1, L2 | |

**Course outcomes:** The students should be able to:
- Identify the AI based problems
- Apply techniques to solve the AI problems
- Define learning and explain various learning techniques
- Discuss on expert systems

**Question paper pattern:**
- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**
<table>
<thead>
<tr>
<th>No.</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.</td>
<td>Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems – Prentice Hal of India.</td>
</tr>
</tbody>
</table>
### Course Learning Objectives:

This course (18CS754) will enable students to:

- Inspect Visual Studio programming environment and toolset designed to build applications for Microsoft Windows
- Understand Object Oriented Programming concepts in C# programming language.
- Interpret Interfaces and define custom interfaces for application.
- Build custom collections and generics in C#
- Construct events and query data using query expressions

### Module – 1

**Introducing Microsoft Visual C# and Microsoft Visual Studio 2015:** Welcome to C#,
Working with variables, operators and expressions, Writing methods and applying scope,
Using decision statements, Using compound assignment and iteration statements, Managing errors and exceptions

**Textbook:** Ch 1 to 6

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
<th><strong>T1:</strong> Chapter 1 – Chapter 6</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>T1:</strong> Chapter 1 – Chapter 6</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RBT: L1, L2</strong></td>
</tr>
</tbody>
</table>

### Module – 2

**Understanding the C# object model:** Creating and Managing classes and objects,
Understanding values and references, Creating value types with enumerations and structures, Using arrays

**Textbook:** Ch 7 to 10

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Textbook:</strong> Ch 7 to 10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RBT: L1, L2</strong></td>
</tr>
</tbody>
</table>

### Module – 3

Understanding parameter arrays, Working with inheritance, Creating interfaces and defining abstract classes, Using garbage collection and resource management

**Textbook:** Ch 11 to 14

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Textbook:</strong> Ch 11 to 14</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RBT: L1, L2</strong></td>
</tr>
</tbody>
</table>

### Module – 4

Defining Extensible Types with C#: Implementing properties to access fields, Using indexers, Introducing generics, Using collections

**Textbook:** Ch 15 to 18

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Textbook:</strong> Ch 15 to 18</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RBT: L1, L2</strong></td>
</tr>
</tbody>
</table>

### Module – 5

Enumerating Collections, Decoupling application logic and handling events, Querying in-memory data by using query expressions, Operator overloading

**Textbook:** Ch 19 to 22

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Textbook:</strong> Ch 19 to 22</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RBT: L1, L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RBT: L1, L2</strong></td>
</tr>
</tbody>
</table>

### Course outcomes:
The students should be able to:

- Build applications on Visual Studio .NET platform by understanding the syntax and semantics of C#
- Demonstrate Object Oriented Programming concepts in C# programming language
- Design custom interfaces for applications and leverage the available built-in interfaces in building complex applications.
- Illustrate the use of generics and collections in C#
- Compose queries to query in-memory data and define own operator behaviour

**Question paper pattern:**

The question paper will have TEN questions.  
There will be TWO questions from each module.  
Each question will have questions covering all the topics under a module.  
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**


**Reference Books:**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CSL76</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>0:0:2</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Lab Contact Hours</td>
<td>36</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

Credits – 2

Course Learning Objectives: This course (18CSL76) will enable students to:
- Implement and evaluate AI and ML algorithms in and Python programming language.

Descriptions (if any):
Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.

Programs List:
1. Implement A* Search algorithm.
2. Implement AO* Search algorithm.
3. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
4. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
5. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
6. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
7. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
8. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
9. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs

Laboratory Outcomes: The student should be able to:
- Implement and demonstrate AI and ML algorithms.
- Evaluate different algorithms.

Conduct of Practical Examination:
- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (*Course to change in accordance with university regulations*)
  - For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
r) For laboratories having PART A and PART B
   i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
   ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks
INTERNET OF THINGS
(Effective from the academic year 2018 -2019)
SEMESTER – VIII

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS81</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

CREDITS –3

Course Learning Objectives: This course (18CS81) will enable students to:

- Assess the genesis and impact of IoT applications and architectures in real world.
- Illustrate diverse methods of deploying smart objects and connect them to network.
- Compare different Application protocols for IoT.
- Infer the role of Data Analytics and Security in IoT.
- Identify sensor technologies for sensing real world entities and understand the role of IoT in various domains of Industry.

Module 1


Textbook 1: Ch.1, 2
RBT: L1, L2, L3

Module 2


Textbook 1: Ch.3, 4
RBT: L1, L2, L3

Module 3


Textbook 1: Ch.5, 6
RBT: L1, L2, L3

Module 4


Textbook 1: Ch.7, 8
RBT: L1, L2, L3

Module 5

### Course Outcomes:
The student will be able to:

- Interpret the impact and challenges posed by IoT networks leading to new architectural models.
- Compare and contrast the deployment of smart objects and the technologies to connect them to network.
- Appraise the role of IoT protocols for efficient network communication.
- Elaborate the need for Data Analytics and Security in IoT.
- Illustrate different sensor technologies for sensing real world entities and identify the applications of IoT in Industry.

### Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:


### Reference Books:


### Mandatory Note:

Distribution of CIE Marks is as follows (Total 40 Marks):

- 20 Marks through IA Tests
- 20 Marks through practical assessment

Maintain a copy of the report for verification during LIC visit.

### Possible list of practicals:

1. Transmit a string using UART
2. Point-to-Point communication of two Motes over the radio frequency.
3. Multi-point to single point communication of Motes over the radio frequency, LAN (Sub-netting).
4. I2C protocol study
5. Reading Temperature and Relative Humidity value from the sensor
# MOBILE COMPUTING
(Effective from the academic year 2018 -2019)

**SEMMETER – VIII**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS821</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS –3**

**Course Learning Objectives:** This course (18CS821) will enable students to:
- Define concepts of wireless communication.
- Compare and contrast propagation methods, Channel models, capacity calculations multiple antennas and multiple user techniques used in the mobile communication.
- Explain CDMA, GSM, Mobile IP, WImax and Different Mobile OS
- Illustrate various Markup Languages CDC, CLDC, MIDP; Programming for CLDC, MIDlet model and security concerns

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>

**Textbook 1:** 2.4 - 2.6, 4.4 - 4.6, 5, 6.
**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>

**Textbook 1:** 7, 9.2 - 9.7, 12.2 - 12.6
**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
</table>

**Textbook 2:** 7, 8.
**RBT:** L1, L2

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Building Wireless Internet Applications: Thin client overview: Architecture, the client, Middleware, messaging Servers, Processing a Wireless request, Wireless Applications Protocol (WAP) Overview, Wireless Languages: Markup Languages, HDML, WML, 10 Hours HTML, cHTML, XHTML, VoiceXML.</td>
<td>08</td>
</tr>
<tr>
<td>Textbook 2: 11, 12, 13</td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td></td>
</tr>
<tr>
<td>RBT: L1, L2</td>
<td></td>
</tr>
</tbody>
</table>

**Module 5**

J2ME: Introduction, CDC, CLDC, MIDP; Programming for CLDC, MIDlet model, Provisioning, MIDlet life-cycle, Creating new application, MIDlet event handling, GUI in MIDP, Low level GUI Components, Multimedia APIs; Communication in MIDP, Security Considerations in MIDP.

**Textbook 1: 15.1 - 15.10**

RBT: L1, L2

**Course Outcomes:** The student will be able to:

- Explain state of art techniques in wireless communication.
- Discover CDMA, GSM, Mobile IP, Wimax
- Demonstrate program for CLDC, MIDlet model and security concerns

**Question paper pattern:**

The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**


**Reference Books:**

## Course Code: 18CS822

### Course Code: 18CS822

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>CIE Marks</th>
<th>SEE Marks</th>
<th>Exam Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>18CS822</td>
<td>Storage Area Networks</td>
<td>40</td>
<td>60</td>
<td>03</td>
</tr>
</tbody>
</table>

**Course Code:** 18CS822  
**Effective from the academic year:** 2018-2019  
**Semester:** VII  
**Course Learning Objectives:** This course (18CS822) will enable students to:

- Evaluate storage architectures,
- Define backup, recovery, disaster recovery, business continuity, and replication
- Examine emerging technologies including IP-SAN
- Understand logical and physical components of a storage infrastructure
- Identify components of managing and monitoring the data center
- Define information security and identify different storage virtualization technologies

### Module 1: Contact Hours

<table>
<thead>
<tr>
<th>Module 1</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Storage System: Introduction to Information Storage:</strong> Information Storage, Evolution of Storage Architecture, Data Center Infrastructure, Virtualization and Cloud Computing. <strong>Data Center Environment:</strong> Application Database Management System (DBMS), Host (Compute), Connectivity, Storage, Disk Drive Components, Disk Drive Performance, Host Access to Data, Direct-Attached Storage, Storage Design Based on Application. <strong>Textbook:</strong> Ch.1.1 to 1.4, Ch.2.1 to 2.10 <strong>RBT:</strong> L1, L2</td>
<td>08</td>
</tr>
</tbody>
</table>

### Module 2: Contact Hours

<table>
<thead>
<tr>
<th>Module 2</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Protection - RAID:</strong> RAID Implementation Methods, RAID Array Components, RAID Techniques, RAID Levels, RAID Impact on Disk Performance, RAID Comparison. <strong>Intelligent Storage Systems:</strong> Components of an Intelligent Storage System, Types of Intelligent Storage Systems. <strong>Fibre Channel Storage Area Networks - Fibre Channel:</strong> Overview, The SAN and Its Evolution, Components of FC SAN. <strong>Textbook:</strong> Ch.3.1 to 3.6, Ch. 4.1, 4.3, Ch. 5.1 to 5.3 <strong>RBT:</strong> L1, L2</td>
<td>08</td>
</tr>
</tbody>
</table>

### Module 3: Contact Hours

<table>
<thead>
<tr>
<th>Module 3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IP SAN and FCoE:</strong> iSCSI, FCIP, Network-Attached Storage: General-Purpose Servers versus NAS Devices, Benefits of NAS, File Systems and Network File Sharing, Components of NAS, NAS I/O Operation, NAS Implementations, NAS File-Sharing Protocols, Factors Affecting NAS Performance <strong>Textbook:</strong> Ch.6.1, 6.2, Ch. 7.1 to 7.8 <strong>RBT:</strong> L1, L2</td>
<td>08</td>
</tr>
</tbody>
</table>

### Module 4: Contact Hours

<table>
<thead>
<tr>
<th>Module 4</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Introduction to Business Continuity:</strong> Information Availability, BC Terminology, BC Planning Life Cycle, Failure Analysis, Business Impact Analysis, BC Technology Solutions, <strong>Backup and Archive:</strong> Backup Purpose, Backup Considerations, Backup Granularity, Recovery Considerations, Backup Methods, Backup Architecture, Backup and Restore Operations, Backup Topologies, Backup in NAS Environments <strong>Textbook:</strong> Ch.9.1 to 9.6, Ch. 10.1 to 10.9 <strong>RBT:</strong> L1, L2</td>
<td>08</td>
</tr>
</tbody>
</table>

### Module 5: Contact Hours

<table>
<thead>
<tr>
<th>Module 5</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Local Replication:</strong> Replication Terminology, Uses of Local Replicas, Replica Consistency, Local Replication Technologies, Tracking Changes to Source and Replica, Restore and Restart Considerations, Creating Multiple Replicas. <strong>Remote Replication:</strong> Modes of Remote</td>
<td>08</td>
</tr>
</tbody>
</table>
### Replication, Remote Replication Technologies


**Textbook1:** Ch.11.1 to 11.7, Ch. 12.1, 12.2, Ch. 14.1 to 14.4

**RBT:** L1, L2

### Course Outcomes:
The student will be able to:

- Identify key challenges in managing information and analyze different storage networking technologies and virtualization
- Explain components and the implementation of NAS
- Describe CAS architecture and types of archives and forms of virtualization
- Illustrate the storage infrastructure and management activities

### Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:


### Reference Books:

## NOSQL DATABASE
(Effective from the academic year 2018-2019)

**SEMESTER – VIII**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>18CS823</th>
<th>CIE Marks</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
<td>SEE Marks</td>
<td>60</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
<td>Exam Hours</td>
<td>03</td>
</tr>
</tbody>
</table>

**CREDITS – 3**

### Course Learning Objectives:
This course (18CS823) will enable students to:

- Define, compare and use the four types of NoSQL Databases (Document-oriented, KeyValue Pairs, Column-oriented and Graph).
- Demonstrate an understanding of the detailed architecture, define objects, load data, query data and performance tune Column-oriented NoSQL databases.
- Explain the detailed architecture, define objects, load data, query data and performance tune Document-oriented NoSQL databases.

### Module 1

<table>
<thead>
<tr>
<th>Why NoSQL? The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, A (mostly) Standard Model, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Aggregate Data Models; Aggregates, Example of Relations and Aggregates, Consequences of Aggregate Orientation, Key-Value and Document Data Models, Column-Family Stores, Summarizing Aggregate-Oriented Databases. More Details on Data Models; Relationships, Graph Databases, Schemaless Databases, Materialized Views, Modeling for Data Access, Textbook1: Chapter 1, 2, 3 RBT: L1, L2, L3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
<td>08</td>
</tr>
</tbody>
</table>

### Module 2

<table>
<thead>
<tr>
<th>Distribution Models; Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication. Consistency, Update Consistency, Read Consistency, Relaxing Consistency, The CAP Theorem, Relaxing Durability, Quorums. Version Stamps, Business and System Transactions, Version Stamps on Multiple Nodes Textbook1: Chapter 4, 5, 6 RBT: L1, L2, L3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
<td>08</td>
</tr>
</tbody>
</table>

### Module 3

<table>
<thead>
<tr>
<th>Map-Reduce, Basic Map-Reduce, Partitioning and Combining, Composing Map-Reduce Calculations, A Two Stage Map-Reduce Example, Incremental Map-Reduce Key-Value Databases, What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preference, Shopping Cart Data, When Not to Use, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets Textbook1: Chapter 7, 8 RBT: L1, L2, L3</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
<td>08</td>
</tr>
</tbody>
</table>

### Module 4

<table>
<thead>
<tr>
<th>Document Databases, What Is a Document Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, E-Commerce Applications, When Not to Use, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
<td>08</td>
</tr>
<tr>
<td>Textbook1: Chapter 9</td>
<td>RBT: L1, L2, L3</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td><strong>Module 5</strong></td>
<td></td>
</tr>
<tr>
<td>Graph Databases, What Is a Graph Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Connected Data, Routing, Dispatch, and Location-Based Services, Recommendation Engines, When Not to Use.</td>
<td>08</td>
</tr>
<tr>
<td>Textbook1: Chapter 11</td>
<td>RBT: L1, L2, L3</td>
</tr>
<tr>
<td><strong>Course Outcomes:</strong> The student will be able to:</td>
<td></td>
</tr>
<tr>
<td>• Define, compare and use the four types of NoSQL Databases (Document-oriented, KeyValue Pairs, Column-oriented and Graph).</td>
<td></td>
</tr>
<tr>
<td>• Demonstrate an understanding of the detailed architecture, define objects, load data, query data and performance tune Column-oriented NoSQL databases.</td>
<td></td>
</tr>
<tr>
<td>• Explain the detailed architecture, define objects, load data, query data and performance tune Document-oriented NoSQL databases.</td>
<td></td>
</tr>
<tr>
<td><strong>Question Paper Pattern:</strong></td>
<td></td>
</tr>
<tr>
<td>• The question paper will have ten questions.</td>
<td></td>
</tr>
<tr>
<td>• Each full Question consisting of 20 marks</td>
<td></td>
</tr>
<tr>
<td>• There will be 2 full questions (with a maximum of four sub questions) from each module.</td>
<td></td>
</tr>
<tr>
<td>• Each full question will have sub questions covering all the topics under a module.</td>
<td></td>
</tr>
<tr>
<td>• The students will have to answer 5 full questions, selecting one full question from each module.</td>
<td></td>
</tr>
<tr>
<td><strong>Textbooks:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Reference Books:</strong></td>
<td></td>
</tr>
<tr>
<td>Course Code</td>
<td>18CS824</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Number of Contact Hours/Week</td>
<td>3:0:0</td>
</tr>
<tr>
<td>Total Number of Contact Hours</td>
<td>40</td>
</tr>
</tbody>
</table>

**CREDITS –3**

**Course Learning Objectives:** This course (18CS824) will enable students to:

- Define technologies of multicore architecture and performance measures
- Demonstrate problems related to multiprocessing
- Illustrate windows threading, posix threads, openmp programming
- Analyze the common problems in parallel programming

<table>
<thead>
<tr>
<th>Module</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Module -1</strong></td>
<td>08</td>
</tr>
</tbody>
</table>
**Textbook 1:** Ch.1, 2  
**RBT:** L1, L2, L3 | 08 |
| **Module -2** | 08 |
**Textbook 1:** Ch.3, 4  
**RBT:** L1, L2, L3 | 08 |
| **Module – 3** | 08 |
| Threading APIs: Threading APIs for Microsoft Windows, Win32/MFC Thread APIs, Threading APIs for Microsoft .NET Framework, Creating Threads, Managing Threads, Thread Pools, Thread Synchronization, POSIX Threads, Creating Threads, Managing Threads, Thread Synchronization, Signaling, Compilation and Linking.  
**Textbook 1:** Ch.5  
**RBT:** L1, L2, L3 | 08 |
| **Module-4** | 08 |
| OpenMP: A Portable Solution for Threading: Challenges in Threading a Loop, Loop-carried Dependence, Data-race Conditions, Managing Shared and Private Data, Loop Scheduling and Portioning, Effective Use of Reductions, Minimizing Threading Overhead, Work-sharing Sections, Performance-oriented Programming, Using Barrier and No wait, Interleaving |
### Module-5


**Textbook 1: Ch.7**  
**RBT: L1, L2, L3**

### Course Outcomes:

- Identify the limitations of ILP and the need for multicore architectures
- Define fundamental concepts of parallel programming and its design issues
- Solve the issues related to multiprocessing and suggest solutions
- Make out the salient features of different multicore architectures and how they exploit parallelism
- Demonstrate the role of OpenMP and programming concept

### Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

### Textbooks:


### Reference Books: