

## ANNEXURE I

### Krishna Kanta Handiqui State Open University Course Curriculum of BCA Programme Credit Distribution for Bachelor of Computer Application

Sl. No.	Nature of Courses	Total no. of Courses	Credit	Total Credits
1	DSC (Discipline Specific Core)	20	4	80
2	Minor (For other Discipline)	8	4	32
3	IDC (Inter-Disciplinary Course)	3	3	9
4	AEC (Ability Enhancement Course)	4	3+3+4+4	14
5	VAC (Value Added Course)	2	3+3	6
6	SEC (Skills Enhancement Courses)	4	3+3+4+4	14
7	Seminar	1	4	4
8	Project	1	12	12
	<b>Total</b>	<b>44</b>		<b>171</b>

### Course Distribution for Bachelor of Computer Applications

Semester	DSC	Minor (For other Discipline)	IDC	AEC	VAC	SEC	Project/ Dissertation / Seminar	Semester wise credits
I	DSC 1	Minor 1	IDC 01	AEC 01	VAC 01	SEC 01		20
II	DSC 2	Minor 2	IDC 02	AEC 02	VAC 02	SEC 02		20
III	DSC 3, DSC 4	Minor 3	IDC 03	AEC 03		SEC 03		23
IV	DSC 5, DSC 6, DSC 7	Minor 4		AEC 04				20
V	DSC 8, DSC 9, DSC 10, DSC 11	Minor 5						20
VI	DSC 12, DSC 13, DSC 14, DSC 15	Minor 6						20
VII	DSC 16, DSC 17, DSC 18	Minor 7				SEC 04	Seminar	24
VIII	DSC 19, DSC 20	Minor 8					Project	24
<b>Total Credits</b>	<b>80</b>	<b>32</b>	<b>9</b>	<b>14</b>	<b>6</b>	<b>14</b>	<b>16</b>	<b>171</b>

## Semester-wise Course and Credit Distribution of BCA

Semester	DSC	Minor (for other Discipline)	IDC	AEC	VAC	SEC	Project/ Seminar	Total Credit
<b>I</b>	DSC 1: Computer Fundamentals and PC Software	Minor 1: Computer Fundamentals	IDC 01	AEC 01 (3)	VAC 01 (3)	SEC 01 (3)		<b>20</b>
<b>II</b>	DSC 2: Digital Techniques	Minor 2: Fundamentals of PC Software	IDC 02	AEC 02 (3)	VAC 02 (3)	SEC 02 (3)		<b>20</b>
<b>Exit 1</b>	<b>UG Certificate Programme</b>							<b>40</b>
<b>III</b>	DSC 3: Programming in C	Minor 3: Fundamentals of Database Management Systems	IDC 03	AEC 03 (4)		SEC 03 (4)		<b>23</b>
	DSC 4: Computer Organization and Architecture							
<b>IV</b>	DSC 5: Data Structure through C Language	Minor 4: Open Source Software		AEC 04 (4)				<b>20</b>
	DSC 6: Object Oriented Programming through C++							
	DSC 7: Data Communication and Computer Network							
<b>Exit 2</b>	<b>UG Diploma Programme</b>							<b>83</b>
<b>V</b>	DSC 8: Database Management Systems	Minor 5: Fundamentals of Web Technology						<b>20</b>
	DSC 9: Operating System							
	DSC 10: Mobile Application Development							
	DSC 11: Linux System Administration							
<b>VI</b>	DSC 12: Computer Graphics	Minor 6: Digital Marketing						<b>20</b>
	DSC 13: Software Engineering							
	DSC 14: Web Technology							
	DSC 15: Introduction to Python Programming							
<b>Exit 3</b>	<b>UG Degree Programme</b>							<b>123</b>
<b>VII</b>	DSC 16: Full Stack Development	Minor 7: Digital Fluency				SEC 04 (4)	Seminar (4)	<b>24</b>
	DSC 17: Introduction to Image Processing							
	DSC 18: Data Mining and Data Warehousing							
<b>VIII</b>	DSC 19: Open Source Software	Minor 8: Fundamentals of Python Programming					Project (12)	<b>24</b>
	DSC 20: Fundamentals of Artificial Intelligence and Machine Learning							
<b>Exit 4</b>	<b>UG Degree Honours/Research Programme</b>							
<b>Total Credit</b>	80	32	9	14	6	14	16	<b>171</b>

## Marks Distribution of Courses for Bachelor of Computer Applications

Sem.	Course	Whether DSC/Minor/ IDC/AE C/ VAC/SEC	Marks	Credit
1 <sup>st</sup>	DSC 1: Computer Fundamentals and PC Software	DSC	20F+30P+50S	4
	Minor 1: Computer Fundamentals	Minor	30F+70S	4
	IDC 01	IDC		3
	AEC 01	AEC		3
	VAC 01	VAC		3
	SEC 01	SEC		3
2 <sup>nd</sup>	DSC 2: Digital Techniques	DSC	30F+70S	4
	Minor 2: Fundamentals of PC Software	Minor	20F+30P+50S	4
	IDC 02	IDC		3
	AEC 02	AEC		3
	VAC 02	VAC		3
	SEC 02	SEC		3
3 <sup>rd</sup>	DSC 3: Programming in C	DSC	20F+30P+50S	4
	DSC 4: Computer Organization and Architecture	DSC	30F+70S	4
	Minor 3: Fundamentals of Database Management Systems	Minor	20F+30P+50S	4
	IDC 03	IDC		3
	AEC 03	AEC		4
	SEC 03	SEC	30F+70S	4
4 <sup>th</sup>	DSC 5: Data Structure through C Language	DSC	20F+30P+50S	4
	DSC 6: Object Oriented Programming through C++	DSC	20F+30P+50S	4
	DSC 7: Data Communication and Computer Network	DSC	30F+70S	4
	Minor 4: Open Source Software	Minor	20F+30P+50S	4
	AEC 04	AEC		4
5 <sup>th</sup>	DSC 8: Database Management Systems	DSC	20F+30P+50S	4
	DSC 9: Operating System	DSC	30F+70S	4
	DSC 10: Mobile Application Development	DSC	20F+30P+50S	4
	DSC 11: Linux System Administration	DSC	30F+70S	4
	Minor 5: Fundamentals of Web Technology	Minor	20F+30P+50S	4
6 <sup>th</sup>	DSC 12: Computer Graphics	DSC	30F+70S	4
	DSC 13: Software Engineering	DSC	30F+70S	4
	DSC 14: Web Technology	DSC	20F+30P+50S	4
	DSC 15: Introduction to Python Programming	DSC	20F+30P+50S	4
	Minor 6: Digital Marketing	Minor	30F+70S	4
7 <sup>th</sup>	DSC 16: Full Stack Development	DSC	20F+30P+50S	4
	DSC 17: Introduction to Image Processing	DSC	20F+30P+50S	4
	DSC 18: Data Mining and Data Warehousing	DSC	30F+70S	4
	Minor 7: Digital Fluency	Minor	20F+30P+50S	4
	SEC 04	SEC	100	4
	Seminar	Seminar/ Presentation	100	4
8 <sup>th</sup>	DSC 19: Open Source Software	DSC	20F+30P+50S	4
	DSC 20: Fundamentals of Artificial Intelligence and	DSC	30F+70S	4

	Machine Learning			
	Minor 8: Fundamentals of Python Programming	Minor	20F+30P+50S	4
	Project	Project	200	12
<b>Total:</b>			<b>4400</b>	<b>171</b>

**N.B.:**

F = Formative Assessment (i.e. Internal Assessment)  
P = Practical  
S = Summative Assessment (i.e. End Term Examination)

1 credit = 30 hours of learning

5 credit = 30 X 4 = 120 hours of learning

\* The detailed syllabus of AEC, VAC, IDC, SEC are listed in ANNEXURE V

**Details of Courses Types to be taken by Learners:**

**Major/ Discipline Specific Core Course (DSC):** Learners have to take compulsory 20 (Twenty) Discipline Specific Core Courses of 4 credits each from of his/her own opted discipline.

**Minor (For other Discipline) Course:** Learners have to take compulsory 8 (Eight) courses of 4 credits each exclusively from one discipline to get a minor specialization in that discipline excluding his/her own discipline. Learners can also opt for compulsory 8 (Eight) courses of 4 credits each from different disciplines excluding his/her own discipline in which case he/she will not get a minor specialization.

**Inter-Disciplinary Course (IDC):** Learners have to take 3 (Three) Multi-Disciplinary Courses of 3 credits each.

**Ability Enhancement Course (AEC):** Learners have to take 4 (Four) Ability Enhancement Courses of either 3 or 4 credit which add up to a sum of 14 credits.

**Value Added Course (VAC):** Learners have to take 2 (Two) Value Added Courses of 3 credit each which add up to a sum of 6 credits.

**Skill Enhancement Course (SEC):** Learners have to select at least 4 (Four) Skill Enhancement Courses of either 3 or 4 credit which add up to a sum of 14 credits. The SEC courses may be from his/her own discipline or other under-graduate disciplines of the University.

**Seminar Presentation/Internship:** Learners will have to either give seminar presentation or carry out internship work in an organization to fulfil this criterion.

**Field Work/ Project/ Internship:** Learners will have to do project work under the guidance of suitable project supervisor, submit a project report and give viva-voce on the project done to fulfil this criterion.

## ANNEXURE II

### Detailed Course-Wise Syllabus of the BCA Programme

#### Syllabus of Discipline Specific Core (DSC) Courses

##### SEMESTER I: COMPUTER FUNDAMENTALS AND PC SOFTWARE

Course Type:	DSC
Number of Credits:	4
Total Marks:	100 (Formative: 20, Practical: 30, Summative: 50)

##### Course Objective:

- To provide learners the knowledge of computers, its different generations and applications
- To provide learners the knowledge of different components of computers and computer memory
- To provide learners the knowledge of using both Microsoft and Linux operating systems
- To familiarize students in usage of LibreOffice Writer, Calc, Impress and Base.

##### Course Outcome:

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

- understand computers, different generation of computers and its applications
- describe the basic components of computers and different parts of computer memory
- use Microsoft operating system and Linux operating system
- use LibreOffice Writer, Calc, Impress and Base

##### Unit 1: Introduction to Computer

Computer- a Definition; Characteristics of Computer; Generations of Computers; Structural Evolution of Computers; File Structure of Computers; Flynn's Classification of Computer Architecture; Applications of Computer: Advantages of Computer

##### Unit 2: Basic Features of Computer Classification

Computer Classification based on Operating Principles: Analog Computers, Digital Computers, Hybrid Computers; Computer Classification based on Applications: General Purpose Computers, Special Purpose Computers; Computer Classification based on Size and Capability: Micro Computers, Mini Computers, Mainframe Computers, Super Computers

##### Unit 3: Basic Components of Computer

Concept of Bit and Byte; Block Diagram of Computer, CPU, ALU, CU; Computer Memory: Primary and Secondary; Input/Output Devices; Buses: Address, Data, Control.

##### Unit 4: Computer Memory and Storage

Memory Representations; Memory Hierarchy; CPU Registers; Primary Memory: Random Access Memory, Read Only Memory; Secondary Memory; Cache Memory; Virtual Memory

##### Unit 5: MS Windows Operating System

Introduction to Windows, Components of Windows, Customizing the Desktop, Files and Folders.

## **Unit 6: LINUX Operating System**

Open Source and Free Software, Advantages and Disadvantages of Linux Operating System, Concept of Path, Basic Linux Commands, File Permission, Text Editor vi.

## **Unit 7: Introduction to LibreOffice**

Overview of LibreOffice software suite and its features, Advantages of using LibreOffice as an alternative to proprietary software, Installation and configuration of LibreOffice on different platforms

## **Unit 8: Getting started with Writer**

Launching Writer and creating a new document, Navigating the Writer interface and understanding the different components of the document window, Entering and formatting text, applying font styles, and working with paragraphs, Saving, closing, and opening existing documents in different formats.

## **Unit 9: Working with templates and styles in Writer**

Creating and using document templates for consistent formatting and layout, Creating and modifying paragraph and character styles for consistent text formatting, Managing styles in the Styles and Formatting dialog box.

## **Unit 10: Formatting documents in Writer**

Adding and formatting images, tables, and other objects in a document, Applying advanced formatting options, such as page borders, columns, and page backgrounds, Using Writer's built-in tools for spell-checking, grammar-checking, and language translation.

## **Unit 11: Introduction to Calc**

Overview of Calc and its features as a spreadsheet program, Launching Calc and creating a new spreadsheet, Navigating the Calc interface and understanding the different components of the worksheet window, Entering and formatting data, applying cell formatting, and working with formulas.

## **Unit 12: Working with functions and formulas in Calc**

Using built-in functions and formulas for common calculations in Calc, Creating custom functions and formulas for complex calculations, Using conditional formatting to highlight data based on certain criteria.

## **Unit 13: Advanced formatting and charting in Calc**

Applying advanced formatting options, such as conditional formatting, data validation, and custom number formats, Creating and modifying charts to visualize data in different ways, Using Calc's built-in tools for filtering, sorting, and analyzing data

## **Unit 14: Introduction to Impress**

Overview of Impress and its features as a presentation program, Launching Impress and creating a new presentation, Navigating the Impress interface and understanding the different components of the slide window, Adding and formatting slides, applying transitions, and customizing presentation settings

### Unit 15: Working with multimedia and animations in Impress

Adding images, audio, and video to a presentation, Creating and modifying animations to enhance visual effects, Using Impress's built-in tools for recording and rehearsing presentations

### Unit 16: Introduction to Base

Overview of Base and its features as a database management tool, Launching Base and creating a new database, Navigating the Base interface and understanding the different components of the database window, creating tables, forms, queries, and reports using Base's built-in tools

#### References:

1. Sinha, P. K., & Sinha, P. (2010). *Computer fundamentals*. BPB publications.
2. Rajaraman, V., & Adabala, N. (2014). *Fundamentals of computers*. PHI Learning Pvt. Ltd.
3. <https://documentation.libreoffice.org/assets/Uploads/Documentation/en/GS7.0/GS70-GettingStarted.pdf>

## SEMESTER II: DIGITAL TECHNIQUES

Course Type:	DSC
Number of Credits:	4
Total Marks:	100 (Formative: 30, Summative: 70)

#### Course Objective:

- To provide learners the knowledge of different numbers systems used in computers
- To provide learners the knowledge of Boolean algebra, logic gates and logic families
- To provide learners the knowledge of arithmetic, sequential and combinational circuits
- To provide learners the knowledge of registers, counters and memory organization.

#### Course Outcome:

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

- describe the different number systems and do binary arithmetic
- understand Boolean algebra, logic gates and logic families
- perform simplification of Boolean expression by Algebraic method and Karnaugh Map
- describe binary arithmetic circuits, sequential circuits and combinational circuits

#### Unit 1: Introduction to Number Systems

Decimal, Binary, Hexadecimal and Octal number system, Number system conversions

#### Unit 2: Binary Arithmetic

Complement: r's and (r-1)'s complement, Binary addition, Binary subtraction, Binary Multiplication, Binary division.

#### Unit 3: Data Representation

Fixed Point representation and Floating point representation

#### Unit 4: Code Conversion

Gray code, BCD, BCD to Excess-3 code conversion technique, ASCII, EBCDIC

#### Unit 5: Boolean Algebra

Introduction, Properties, De-Morgan's Theorem, Boolean Identities

#### Unit 6: Logic Gates

Logic Gates: AND, OR, NOT, NAND, NOR, XOR; Conversion of the logic gates

#### Unit 7: Floating Point Number Representation

Floating point number, Normalization of floating point, overflow and underflow, detection of overflow, IEEE floating point standard

#### **Unit 8: Logic Families**

Introduction, Resistor Transistor Logic(RTL), Integrated Injection logic(IIL), Diode- Transistor Logic(DTL), Emitter-Coupled Logic(ECL), Transistor- Transistor Logic(TTL), TTL-NAND, Tri State Logic, MOS devices, Logic gates with MOSFET's

#### **Unit 9: Reduction Techniques**

Boolean Expression, SOP, POS, Conversion of SOP to POS and vice-versa, Simplification of Boolean expression by Algebraic method and Karnaugh Map, Don't care condition

#### **Unit 10: Binary Arithmetic Circuits**

Binary Half adder, Full adder, Binary Half subtractor, Full subtractor, 1's complement of n bit adder/subtractor

#### **Unit 11: Combinational circuits**

Multiplexer, 4-to-1 multiplexer, Demultiplexer, Encoder, Priority encoder, Decoder, 3-to-8 Decoder.

#### **Unit 12: Sequential circuits**

Synchronous and Asynchronous Sequential circuit, Flip-Flop and its different types: RS, JK; Master-Slave JK Flip-Flop.

#### **Unit 13: Registers**

Shift register, SISO, PISO, SIPO, PIPO shift register, Application of Shift Register.

#### **Unit 14: Counters**

Synchronous and Asynchronous Counter, Limitations of asynchronous counter, Frequency division counter.

#### **Unit 15: Memory Organization**

Memory types, Random access memory, Access time, RAM package, Static RSM, CMOS memory cell, Dynamic RAM, Read Only Memory, Memory expansion.

#### **References:**

1. Mano, M. M. (2017). *Digital logic and computer design*. Pearson Education India.
2. Talukdar, P. (2004). *Digital Techniques*. N L Publications

### **SEMESTER III: PROGRAMMING IN C**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

#### **Course Objective:**

- To provide learners the concepts of C programming
- To help learners develop logical skills to solve complex problems with C programming
- To familiarize learners with basic concepts of programming so that they can write programs.
- To equip learners with skills to write C programs where concepts of functions, pointers, structures and file handling are used.

#### **Course Outcome:**

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

- describe concepts like pseudocode, algorithm, flowchart and program.
- define data types, variables, constants and keywords, operators and expressions
- design C programs where conditional, loop control statements, arrays and strings are used
- write C programs using, functions, pointers, structures and union



**Unit 1: Introductory Concepts**

Algorithm, Flowchart, Symbols of Flowchart, Pseudo Code, Examples of Algorithm, Flowchart and Pseudocode.

**Unit 2: Elements of C Programming**

Characters used in C, Identifiers, Keywords, Tokens, Constants, Variables.

**Unit 3: Variables and Data types**

Integer, Character, Floating point etc; Initialization of variable during declarations; Symbolic Constants.

**Unit 4: Operators and Expressions**

Expression in C, Different types of operators: Arithmetic, Relational and Logical, Assignment, Conditional, Increment and decrement, Bitwise, Comma and sizeof etc.; Precedence and associativity of operators, type conversion.

**Unit 5: Preprocessor Directives and I/O Functions**

Header Files (stdio, conio), Formatted Input / Output Functions (scanf, printf), Escape Sequences, Character Input/Output Functions (getch, getchar, putchar, gets, puts, getche, clrscr)

**Unit 6: Conditional Statements**

Conditional Statement- if, if- else, nested if-else, switch-case; break, continue, goto.

**Unit 7: Loop Control Structures**

Concept of Loops, Types of loop: while, do-while, for; nested loops

**Unit 8: Storage Class**

Automatic, External, Static, Register, Scope and lifetime of variables, Macro, Preprocessor directive

**Unit 9: Arrays**

Array, Array Declaration, 1-Dimensional array, 2-Dimensional array

**Unit 10: Strings**

String, String Handling Functions: strlen (), strcmp (), strcpy (), strrev (), strcat (), etc.

**Unit 11: Functions**

Function, Function declaration, Function definition, Function call, Formal and Actual parameter, Recursive function.

**Unit 12: Pointers**

Declaring Pointer Variable; Pointer Arithmetic, Pointers and One-Dimensional Arrays, Pointers and Character Arrays, Pointer declaration, Passing Pointer to a Functions as Argument, Dynamic Memory Allocation: malloc (), calloc (), realloc (), free ().

**Unit 13: Structures and Union**

Structure Declarations, Definitions, Defining your typedef, Array of Structure, Pointer to Structure. Union Declaration, Definition, Declaration, Uses

**Unit 14: File Handling**

Concept of File, File Pointer, File Opening in various modes, closing a file, reading and writing on files, Formatted Input / Output, fseek (), ftell (), rewind ().

**References:**

1. Schildt, H. (2021). C the complete reference.
2. Balagurusamy, E. (2016). Programming In Ansi C.
3. Kanetkar, Y. (2018). *Let us C*. BPB publications.

## SEMESTER III: COMPUTER ORGANIZATION AND ARCHITECTURE

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 30, Summative: 70)</b>

### Course Objective:

- To provide learners the knowledge of basic organization of computer
- To provide learners the knowledge of instruction sets, addressing modes and I/O techniques
- To familiarize learners with basic concepts like cache memory, virtual memory and mapping functions
- To provide learners knowledge of external memory, CISC and RISC architectures.

### Course Outcome:

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

- describe the basic organization of the computer
- describe the different types of instruction sets and addressing modes
- explain how the different I/O techniques work
- describe parallel processing, CISC and RISC architecture

### Unit 1: Basic Organization of the Computer

Basic organization of the computer and block level description of the functional Units from program execution point of view; Fetch, decode and execute cycle

### Unit 2: Digital Components

Flip-Flops, Counter, Register, Half adder, Full Adder, Half subtractor, Full subtractor, Coder, Decoder, Multiplexer, De-multiplexer, Magnitude Comparator

### Unit 3: Data Representation

Data representation, computer arithmetic and their implementation; control and data path, data path components, design of ALU and data path, control Unit design

### Unit 4: Computer Arithmetic

Integer representation, sign magnitude representation, twos complement representation, integer arithmetic

### Unit 5: Instruction Sets

Elements of a machine instruction, instruction representation, Simple instruction format, Instruction types, number of addresses, Types of operands, Types of operations. Different Instruction Formats, Instruction Types, Instruction Execution, Assembly language notation

### Unit 6: Addressing Modes

Addressing, Immediate, Direct, Indirect, Registrar, Registrar indirect, Relative

### Unit 7: Input-Output Organization

Different I/O techniques (Programmed I/O, Interrupt-Driven I/O), DMA (Direct Memory Access), I/O Processors.

### Unit 8: Introduction to Cache and Virtual Memory

Memory Hierarchy, Semiconductor memories, internal organization of typical RAM and ROM Memory, Switches, Cache memory, Cache memory access techniques; Mapping functions, Virtual memory, Locality of reference, Paging, Cache Coherence Problem

### Unit 9: Memory and I/O access

Memory Read Write operations, Concept of handshaking, Polling Techniques (Serial and Half Polling) and Interrupt driven I/O, Priority and Daisy Chaining Technique, Introduction to Memory Mapping, Basic concept of Cache and Virtual Memory

### Unit 10: Memory and I/O Interfacing

I/O processor, Priority Encoder, Device Scheduler, Interfacing with the I/O Devices, keyboard,

printer and display interfaces

#### **Unit 11: External Memory**

Magnetic Disk, Magnetic read write operation, sector, track, inter track and inter sector gap, cylinder, fixed head disk, seek time, access time, transfer time, rotational delay, RAID, Optical memory, Magnetic tap

#### **Unit 12: Processor Structure and Function**

Processor organization: Fetch instruction, interpret instruction, fetch data, process data, write data, Registrar organization, Control and status registrar, Instruction cycle, Instruction pipelining (definition only)

#### **Unit 13: Introduction to Parallel Processing**

Introduction to Pipelining and Basics of Parallel Processing, Scalable Architecture

#### **Unit 14: CISC and RISC Architecture**

CISC (Complex Instruction Set Computers), RISC (Reduced Instruction Set Computers), Examples of CISC and RISC.

#### **References:**

1. Mano, M. M. (1993). *Computer system architecture*. Prentice-Hall, Inc..
2. Stallings, W. (2003). *Computer organization and architecture: designing for performance*. Pearson Education India.

### **SEMESTER IV: DATA STRUCTURE THROUGH C LANGUAGE**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

#### **Course Objective:**

- To provide learners knowledge of data structures and algorithm complexities
- To provide learners knowledge of data structures like arrays, linked list, stacks and queues
- To provide learners knowledge of different searching and sorting algorithms
- To provide learners knowledge of trees, binary search trees, graphs, tree and graph traversal algorithms

#### **Course Outcome:**

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

- describe basic concepts of data structures and algorithm complexity
- design programs where concept of arrays, linked list, stacks and queues are used
- write C programs for different search techniques and sorting algorithms
- explain the concepts related to trees, binary trees and graphs

#### **Unit 1: Introduction to Data Structure**

Basic concept of data, Data type, Abstract Data Type, Types of Data Structure, Dynamic Memory allocation.

#### **Unit 2: Algorithms**

Algorithm, Complexity, Time-Space Complexity, Algorithmic Notation

#### **Unit 3: Arrays**

Arrays: Types, One dimensional array, Multidimensional arrays, memory representation, address translation functions, Applications of array, Random accessing of array element

#### **Unit 4: Operations on Arrays**

Traversal, Search, Problems associated with Insertion and Deletion, Sorting, Reversing, Merging

#### **Unit 5: Recursion**

Function, Rules for Recursive Function, Recursion versus Iteration, Advantages and Disadvantages

#### **Unit 6: Linked List**

Introduction to Linked List, Important terms associated with Linked list, Representation of Single linked list using Static and Dynamic Memory Allocation. Circular linked list, doubly linked list, Applications of Linked List.

#### **Unit 7: Operations on Linked List**

Insertion of node in a linked list (at Start, at End, at a Given Position), deletion of node in a linked list (at Start, at End, at a Given Position), searching and traversal of elements and their comparative studies with implementations using array structure, reversing a singly linked list.

#### **Unit 8: Stacks**

Definitions, LIFO, Representation using array and linked list structure, Applications of stack.

#### **Unit 9: Queues**

Definitions, FIFO, Representation using array, linked representation of queues, application of queue

#### **Unit 10: Searching**

Linear and binary search algorithms, performance and complexity using big 'O' notation

#### **Unit 11: Sorting**

Sorting algorithms (Complexity, advantages and disadvantage, implementation), bubble sort, insertion sort, selection sort, quick sort.

#### **Unit 12: Trees**

Definition, Types of trees, Implementation: Binary Tree, Tree traversal algorithms (in-order, preorder, post-order), infix, postfix, prefix notations

#### **Unit 13: Binary Search Tree**

Searching in Binary Search Tree (BST), insertion and deletion in Binary Search Trees, Height of BST, B-Tree, B Tree.

#### **Unit 14: Graphs**

Introduction to Graphs, Edge, Vertex, Depth first search (DFS) and Breadth first search (BFS) technique

#### **References:**

1. Langsam, Y., Augenstein, M. J., & Tenenbaum, A. M. (1996). *Data Structures using C and C++*. Prentice Hall Press.
2. Hopcroft, J. E., Ullman, J. D., & Aho, A. V. (1983). *Data structures and algorithms* (Vol. 175). Boston, MA, USA:: Addison-wesley.
3. Horowitz, E., & Sahni, S. (1982). *Fundamentals of data structures*.

### **SEMESTER IV: OBJECT ORIENTED PROGRAMMING THROUGH C++**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

#### **Course Objective:**

- To provide learners the knowledge of concepts related to OOP like abstraction, encapsulation, inheritance and polymorphism
- To equip learners to design C++ programs using classes and objects, constructors and destructors
- To provide learners knowledge of concepts like operator overloading, virtual functions and polymorphism
- To familiarize learners with learners on how to handle and use files and templates in C++

#### **Course Outcome:**

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

- design C++ programs using classes and objects, constructors and destructors

- write C++ programs for operator overloading and different types of inheritances
- describe the concept of virtual functions and polymorphism
- describe the concept of file handling, class and function templates

### **UNIT 1: Introduction to Object-Oriented Programming**

Basic concept of OOP, Comparison of Procedural Programming and OOP, Benefits of OOP, C++ compilation, Difference between C and C++.

### **UNIT 2: Features of Object-Oriented Programming**

Abstraction, Encapsulation, Inheritance, Polymorphism

### **UNIT 3: Elements of C++ Language**

Tokens and identifiers: Character set and symbols, Keywords, C++ identifiers. Variables and constants: Integers & Characters, Constants and Symbolic constants, Dynamic initialization of variables, Reference variables, Basic data types in C++.

### **UNIT 4: Operators and Manipulators**

Operators, Types of Operators in C++, Precedence and Associativity, Manipulators

### **UNIT 5: Decision and Control Structures**

If statement, If-else statement, switch statements, Loop: while, do-while, for; Jump statements: break, continue, Goto.

### **UNIT 6: Array, Pointers and Structure**

Arrays, pointer, structure, unions;

### **UNIT 7: Functions in C++**

The main() function, function prototype, function call, function definition, function parameter, inline function, default arguments, function overloading, friend function

### **UNIT 8: Introduction to Classes and Objects**

Classes in C++, class declaration, declaring objects, Defining Member functions, Inline member function, Array of objects, Objects as function argument, Static data member and member function, friend class

### **UNIT 9: Constructors and Destructors**

Constructors, Instantiation of objects, Default constructor, Parameterized constructor, Copy constructor and its use, Destructors, Constraints on constructors and destructors, Dynamic initialization of objects, Constructor Overloading.

### **UNIT 10: Operator Overloading**

Overloading unary operators: Operator keyword, arguments and return value; Overloading Unary and binary operators: arithmetic operators, manipulation of strings using operators, Type conversions.

### **UNIT 11: Inheritance**

Derived class and base class: Defining a derived class, Accessing the base class member, Inheritance: multilevel, multiple, hierarchical, hybrid; Virtual base class, Abstract class.

### **UNIT 12: Virtual Functions and Polymorphism**

Virtual functions, Pure virtual functions; Polymorphism, Categorization of polymorphism techniques: Compile time polymorphism, Runtime polymorphism.

### **UNIT 13: File Handling**

File classes, Opening and Closing a file, File modes, Manipulation of file pointers, Functions for I/O operations, File handling functions.

### **Unit 14: Templates**

Introduction to Template, Function Template, Function Template Overloading, Class Template, Default Template Arguments.

### **References Books:**

1. Schildt, H. (2003). *C++: The complete reference*. McGraw-Hill.
2. Stroustrup, B. (2013). *The C++ programming language*. Pearson Education.
3. Balagurusamy, E. (2010). *Object Oriented Programming with C++* | Tata McGraw Hill Education Pvt. Ltd.,.

## SEMESTER IV: DATA COMMUNICATION AND COMPUTER NETWORK

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 30, Summative: 70)</b>

### Course Objective:

- To provide learners knowledge of various concepts in data communication and computer networks.
- To provide learners knowledge of different layers in the OSI and TCP/IP model
- To provide learners knowledge of TCP and UDP protocols
- To provide learners knowledge of network standard protocols like FTP, Telnet etc.

### Course Outcome:

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

- describe the different types of networks and network topologies.
- explain OSI and TCP/IP reference models and the functions of all the layers.
- explain the fundamentals of standard protocols like TCP, UDP, FTP, Telnet etc.
- describe how different network devices and Ethernet works

### Unit 1: Introduction to Data Communication

Introduction, Signals, Types of Signal: Analog, Digital, Block representation of Data Communication System, Definition of Protocols and Standards, Channel, Bandwidth, SNR

### Unit 2: Types of Data Communication

Data Communication, Parallel and Serial Communication, Modes of Communication: Simplex, Half Duplex, Full Duplex; Asynchronous and Synchronous Communication

### Unit 3: Digital Transmission Fundamentals

Digital Signal (PCM, Delta Modulation, Adaptive Delta Modulation, ASK, PSK, FSK), Multiplexing and De-multiplexing, Detection and Correction of Transmission Errors (Parity and Hamming Code Technique), Data Compression and Encryption, Concept of BCH Coding

### Unit 4: Introduction to Computer Networks

Computer Network: Definition, Goals, Structure; Broadcast and Point-To-Point Networks; Types of Networks (LAN, MAN, WAN), Network Topologies, Network Protocols and Standards, Networking Switching Techniques

### Unit 5: Network Models

Network Reference Models, OSI Reference Model, TCP/IP reference Model

### Unit 6: Physical Layer

Physical Layer, Transmission Media, Guided and Unguided Media, Repeaters, Hub

### Unit 7: Data Link Layer

Data Link Layer, Flow Control, Access Protocol, Bridges and Switches, MAC, LLC, Frame, Frame Format

### Unit 8: Network Layer

Network Layer, Routing Protocols, Internet Protocol, IP Addresses, Sub-netting

### Unit 9: Transport Layer

Process-To-Process delivery, User Datagram Protocol, Transmission Control Protocol: Characteristics, TCP Segment, Connection Establishment and termination

### Unit 10: Session Layer

Interhost communication; Services- Dialog control, Token management, Synchronization; Protocols- Remote Procedure Call (RPC), NETBIOS.

### Unit 11: Presentation Layer

Services - Data conversion, Translation, Encryption, Compression; Protocols - FTP, Telnet.

### Unit 12: Application Layer

Client-Server Model: Client-Server model, Socket interface; A brief introduction to DNS, SMTP, FTP

### Unit 13: Fundamentals of TCP and UDP

Connectionless and Connection Oriented Protocol, Transmission Control Protocol (TCP), User Datagram Protocol (UDP), IDP

**Unit 14: Network Standard Protocol**

FTP, Telnet, HTTP, Email Protocol, POP, SMTP, Browser, WWW, PING, Remote Login, Socket

**Unit 15: Network Devices**

Gateway, Bridge, Hub, Switch, RS-32C, Network Interface Card (NIC), Network Adapters, Components of NIC, Functions of NIC, Types of NIC; Ethernet: Basic Features, Types of Ethernet, Different Framer Format: IEEE 802.3, IEEE 802.4, IEEE 802.5.

**References Books:**

1. Forouzan, B. A. (2007). *Data communications and networking*. Huga Media.
2. Stallings, W. (2007). *Data and computer communications*. Pearson Education India.
3. Comer, D. E., & Stevens, D. L. (1993). *Internetworking with TCP/IP Vol. III: Client-server programming and applications*. Prentice-Hall, Inc..
4. Bertsekas, D., & Gallager, R. (2021). *Data networks*. Athena Scientific.

**SEMESTER V: DATABASE MANAGEMENT SYSTEMS**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

**Course Objective:**

- To provide learners knowledge of database architecture, data models and relational model
- To provide learners knowledge of functional dependencies and normalization
- To familiarize learners with SQL programming to create, alter and delete data and tables
- To provide learners knowledge of using GUI and connecting to a database

**Course Outcome:**

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

- describe the database architecture, data models and relational model
- explain functional dependencies, normalization and the different types of keys
- use SQL to create, modify, delete tables and databases and write queries and subqueries
- use WAMP and phpMyAdmin

**Unit 1: File Structure**

Data and Information, Concept of Field, Key field; Records and its types, Fixed length records and Variable length records; Files, Operation on files, Primary file organization

**Unit 2: Database System**

Traditional file approach versus Database approach; Data independence, Database System, Database Architecture, The three level of architecture, Mapping, Database administrator, Database Management system, Types of DBMS, Merits and demerits of DBMS

**Unit 3: Data Models**

Conceptual model, Logical model, Physical model, Entity-Relationship (ER) model as a tool for conceptual design: Entities, Attributes and Relationships, Weak and Strong entities, Conversion of ER model into Relational schema, ER modelling symbols.

**Unit 4: The Relational Model**

Relational data model concepts, Integrity constraints: Entity integrity, Referential integrity, Domain Constraints.

**Unit 5: Keys**

Concept of keys, Composite key, Candidate key, Primary key, Alternate key, Foreign key, Defining Primary and Foreign keys in Database.

### Unit 6: Relational Database Design

Database Design, Decomposition, Universal Relation, Functional Dependencies, Prime and Non-prime attributes.

### Unit 7: Normalization

Normalization, First Normal form (1NF), Second Normal form (2NF), Third Normal form (3NF), Boyce-Codd Normal form (BCNF)

### Unit 8: Introduction to SQL

Structured Query Language (SQL), Characteristics of SQL, Advantages of SQL, SQL Data types, Types of SQL commands, DDL, DML, Creating a Database, Creating, Altering, and Deleting Tables

### Unit 9: Working with SQL Part I

Inserting new data, Updating data (the WHERE Clause, The Logical AND and OR operator), Deleting Data.

### Unit 10: Working with SQL Part II

The SELECT statement, Filtering results with the WHERE Clause; Logical Operators and Operator Precedence (Using logical operator NOT, BETWEEN, LIKE, IN); Ordering Results with ORDER BY (Joining columns-concatenation); Selecting data from more than one table.

### Unit 11: Working with SQL Part III

Join, Types of Join, SQL commands: Select... From...Where... Group by ..... Having... Order by..., Tables, Queries, Sub Queries, Insert, Update and Delete operations, Constraints considers (NOT NULL, UNIQUE, Check Primary key, Foreign key)

### Unit 12: Working with GUI

Introduction to WAMP, Installation of WAMP, Working with phpMyAdmin.

### Unit 13: Database Recovery

Concept of database recovery, Backup of database, Types of database failure, Types of database recovery.

### Unit 14: Database Security

Goals of database security, Discretionary Access Control, Mandatory Access control, Statistical Databases, Data Encryption.

### Reference Books:

1. Elmasri, R., Navathe, S. B., Elmasri, R., & Navathe, S. B. (2000). *Fundamentals of Database Systems*. Addison-Wesley.
2. Ramakrishnan, R., Gehrke, J., & Gehrke, J. (2003). *Database management systems* (Vol. 3). New York: McGraw-Hill.
3. Silberschatz, A., Korth, H. F., & Sudarshan, S. (2011). *Database system concepts*.
4. Schwartz, B., Zaitsev, P., & Tkachenko, V. (2012). *High performance MySQL: optimization, backups, and replication*. " O'Reilly Media, Inc."

## SEMESTER V: OPERATING SYSTEM

Course Type:	DSC
Number of Credits:	4
Total Marks:	100 (Formative: 30, Summative: 70)

### Course Objective:

- To provide learners knowledge of concepts related to operating system like process, threads and deadlocks
- To familiarize learners with scheduling algorithms like FCFS, SJF and RR
- To provide learners knowledge of different memory management techniques like paging
- To provide learners knowledge of file systems and I/O management.



## Course Outcome:

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

- explain concepts like process, threads, race condition, mutual exclusion and deadlocks
- describe how different types of scheduling algorithms like FCFS, SJF and RR work
- describe different memory management techniques including segmentation and paging
- explain how file system, I/O management and multiprocessor systems work

### Unit 1: Introduction to Operating System

Operating system, Evolution of Operating Systems- batch system, multi-programmed system, time-sharing system, personal computer system, parallel systems.

### Unit 2: Processes

Concept of process, Process life cycle, Process Control Block (PCB), Process Management

### Unit 3: Threads

Concept of Threads, Uses of Thread, Design Issues of Thread, Thread Management

### Unit 4: Interprocess Communication

Interprocess Communication, Message passing model, shared memory model

### Unit 5: Process synchronization and Coordination

Concept of race conditions, critical-sections, mutual exclusion, Solution to critical section problem, disabling interrupt, Peterson's solution, Sleep & Wake up, The Producer Consumer Problem

### Unit 6: Semaphores

Semaphores, Wait and Signal, Binary Semaphores, Application of semaphores

### Unit 7: Scheduling

Basic concepts, primitive and non-primitive scheduling, Types of scheduling - batch, interactive and real-time, Scheduling criteria, Scheduling algorithms, first come first serve (FCFS), shortest job first (SJF) and round robin (RR) scheduling

### Unit 8: Deadlocks

What is deadlock, Conditions for deadlock, Deadlock detection, Resource allocation graph, Methods for handling deadlocks-ignore problem, recovery & prevention, deadlock avoidance (banker's algorithm), Congestion versus Deadlock

### Unit 9: Memory Management

Memory management requirements, Memory portioning, Fixed and Variable size partitioning, Swapping, Virtual Memory.

### Unit 10: Segmentation and Paging

Basic Concept of Segmentation, Segmentation Architecture, Address Translation in Segmentation, Basic Concepts of Paging, Page Tables, Address Translation in Paging, Logical and Physical Address, Page replacement algorithms (First in First Out, Optimal, LRU).

### Unit 11: File System

File Structure, File naming, File types(directory, regular, device), Sequential access and Random access files, File attributes, Operations on file, Directory Structure- Hierarchical directory structure, Path name (relative and absolute), Operation on directories, File System Implementation Techniques.

### Unit 12: I/O Management

Basic principles of I/O Hardware, I/O Devices, Device controllers, DMA, Principles of I/O Software, Interrupt Handlers, Device Drivers, Device Independent I/O Software

### Unit 13: Security and Protection

Security threats and goals, Authentication, Protection and Access control, Formal model of protection, Cryptography.

### Unit 14: Introduction to Multiprocessor Systems

Introduction to Multiprocessor System, Types of Multiprocessor System, Functional characteristics of Multiprocessor OS, Multiprocessor Synchronization

## References:

1. Silberschatz, A., Peterson, J. L., & Galvin, P. B. (1991). *Operating system concepts*. Addison-Wesley Longman Publishing Co., Inc..
2. Tanenbaum, A. S., *Modern operating systems*. (2007). Pearson Education

3. Nutt, G. J. (2001). *Operating Systems: A Modern Perspective, Lab Update*. Addison-Wesley Longman Publishing Co., Inc..
4. Stallings, W. (1998). *Operating systems internals and design principles*. Prentice-Hall, Inc..

## SEMESTER V: MOBILE APPLICATION DEVELOPMENT

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

### Course Objective:

- To introduce students to the Android platform and its history, the Open Handset Alliance, and the architecture of the Android operating system.
- To provide students with the necessary knowledge and skills to install and use Android Studio for Android software development, including creating a new Android project, defining the project and SDK settings, and customizing the Android Studio IDE components.
- To enable students to design and develop Android applications, including understanding the components of the Android manifest file, creating activities, services, and intents, and managing permissions and application resources.
- To teach students how to create user interfaces in Android applications, including working with basic user interface screen elements, designing user interfaces with layouts, and using drawing and animation features in Android. Additionally, students will learn how to work with data storage options, file I/O, preferences, SQLite databases, content providers, network data access, and web services in Android applications.

### Course Outcome:

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

- describe concepts related to android platform and android studio
- explain how to build an android application
- describe android design essentials like activities, services, intents, permissions and application resources.
- design user interfaces with layouts

### Unit 1: The Android Platform

History of mobile application development, The Open Handset Alliance (OHA), The Android Platform, Android Versions, Native Android Applications, Android SDK Features, Android Architecture, Factors that affect Mobile Application development.

#### Unit 2: Installing Android Studio

System Requirements for Android Studio, Downloading Android Studio, Downloading JDK, Installing JDK, Installing Android Studio, Launching Android Studio.

#### Unit 3: Android Studio for Android Software Development

Features of Android Studio, App Workflow, Android Virtual Devices (AVD), Using Hardware Device to test Application, Android Studio IDE Components, Android Studio Code Editor Customization.

#### Unit 4: Building a sample Android application using Android Studio

Building a sample Android application using Android Studio, Creating a new Android Project, Defining the Project and SDK Settings, Creating Activity

#### Unit 5: Android Project Structure

Android Project Structure, Android Project Files, Android Application Modules, Types of Modules, Project structure settings, Anatomy of an Android Application, Important Android Terminology, Basic Android API Packages, Android Advanced API Packages

**Unit 6: Android Manifest File and its common settings**

Components of Manifest file, Package name and application ID, App components, Permissions, Device compatibility, File conventions, Manifest elements reference.

**Unit 7: Android Application Design Essentials: Activities**

What is activity?, Configuring the AndroidManifest.xml file, Life Cycle of an Activity, Understanding Life Cycle of an Activity, Context, Activity Transition.

**Unit 8: Android Application Design Essentials: Services**

Use of services, Creating a service, Start and Stop Service, Service Life Cycle, Creating your own Service.

**Unit 9: Android Application Design Essentials: Intents**

Intent Structure, Other Operations on Intent, Types of Intent, Intent Resolution, Example of Intent, Explanation of Example, Standard Activity Actions, Standard Broadcast Actions.

**Unit 10: Android Application Design Essentials: Permissions**

Permission approval, Request prompts for dangerous permissions, Permission for optional hardware features, Custom App Permission, Permission Protection Level, Viewing App permission.

**Unit 11: Android Application Design Essentials: Application Resources**

Application Resources, Resource Directory Hierarchy, Resource Value Types, Storing Different Resource Value Types, Accessing Resource Programmatically, Referencing System Resources.

**Unit 12: Basic User Interface Screen elements**

Introduction to Views, Controls and Layout, TextView, EditText, AutoCompleteTextView, Spinner, Buttons, Check Boxes, Radio Groups, Pickers

**Unit 13: Designing User Interfaces with Layouts**

Creating Layouts Using XML Resources, Creating Layouts Programmatically, Built-In Layouts, Frame Layout, Linear Layout, Relative Layout, Table Layout, Data Driven Container

**Unit 14: Drawing and Working with Animation**

Canvas and Paints, Bitmaps, Shapes, Frame by Frame animation, Tweened Animation

**References:**

1. Gargenta, M. (2011). *Learning android*. " O'Reilly Media, Inc."
2. Ableson, F., King, C., & Ortiz, C. E. (2011). *Android in action*. Simon and Schuster.
3. Haseman, C. (2009). *Android essentials*. Apress.

**SEMESTER V: LINUX SYSTEM ADMINISTRATION**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

**Course Objective:**

- To provide learners knowledge of system administrator roles and qualities
- To provide learners knowledge of Linux kernel and file system
- To familiarize learners with learners to use shell programming to write small programs
- To provide learners knowledge of networking protocols and various network security issues.

**Course Outcome:**

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

- describe the role and qualities of a system administrator
- describe Linux kernel and the Linux file system
- use various Linux file handling and filter commands
- describe various networking protocols and basic network security issues

**Unit 1: Introduction to System Administration**

Introduction to System Administration, Role and power of System Administrator, Qualities of good System Administrator

**Unit 2: Introduction to LINUX Operating System**

Basic Features of the Linux operating system. A brief Overview of the most popular Linux Distributions – (Red Hat Enterprise Linux (RHEL), Ubuntu, Debian, Fedora, SUSE)

**Unit 3: Installation of LINUX Operating System**

Installation Requirements, Partitioning the Hard drive in Linux, Installing the Linux system, Installing and Configuring software in Linux

**Unit 4: LINUX Kernel**

Linux kernel and device drivers, System Startup and Shutdown, Standard I/O, Standard error, Redirection and Piping.

**Unit 5: Basics of LINUX File System**

Basics of Linux file system - File system types (ext3, ext4, xfs, jfs, ReiserFS, iso9660 etc.), three basic types of files (ordinary or regular, special or device and directory). I-nodes and file attributes

**Unit 6: File organization**

Absolute and Relative path names, File system Mounting and Unmounting, Organization of the file tree, Standard directories and their contents

**Unit 7: LINUX File Handling Commands**

Files and Directory handling Commands - ls, cd, cp, mv, rm, mkdir, rmdir, Commands for Creating and Viewing ordinary files – cat, more, pg

**Unit 8: LINUX Filter Commands**

Filter Commands – wc, head, tail, cut, tr, grep (with regular expressions). Setting user and group ownership of files and Access permissions – chmod, chown, chgrp commands

**Unit 9: LINUX Shells**

Study of different Linux Shells (sh, bash, csh, zsh), Environment variables, Shell script basics (examples of some simple shell programming)

**Unit 10: Process**

Basic commands for starting and stopping processes, Basic process attributes and their role in Access control. Examining the list of running processes on the system and understand the data presented there. Background process, Job control, Crontab file format, Backup and Restore procedure

**Unit 11: Printing Jobs**

Submit a print job, check the status of a print job, cancel a print job, configuring the Print Queue, Selecting the Print Driver, Editing the Printer configuration

**Unit 12: LINUX Users**

Understanding the “root” account, Becoming a Superuser (su), A limited su (sudo) Managing user accounts - Adding a new user, Modifying and Removing User accounts, Changing Password, System monitoring and logging, Monitoring memory usage, disk space usage and I/O activity .

**Unit 13: Networking in LINUX**

The rules governing IP address classes and netmasks, Network Address, Netmask and Gateway configuring Interface with ifconfig, ping, netstat, traceroute, telnet, understanding the significance of the /etc/services file and well known port numbers

**Unit 14: Network Protocols**

Basics of configuring NFS, NIS, DNS, FTP, Squid Proxy, DHCP server, ip tables and firewall

**Unit 15: Basic Network Security Issues**

Introduction to Basic Network Security Issues, Packet Sniffers, DOS Attacks, Linux Kernel Firewalling, Virtual Private Networks

**References:**

1. Nemeth, E., Snyder, G., & Hein, T. R. (2006). *Linux administration handbook*. Addison-Wesley Professional.
2. Frisch, A. (2002). *Essential system administration: Tools and techniques for linux and unix administration*. " O'Reilly Media, Inc."

## SEMESTER VI: COMPUTER GRAPHICS

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 30, Summative: 70)</b>

### Course Objective:

- To provide learners the basic concepts used in computer graphics.
- To provide learners the knowledge of various graphics systems.
- To implement various algorithms to draw, scan, convert the basic geometrical primitives, transformations, area filling, clipping etc.
- To provide the concept of graphic transformations.
- To provide the concept of viewing and projections.

### Course Outcome:

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

- understand the basics of computer graphics and different graphics systems.
- implement various line, circle, ellipse drawing algorithms.
- use geometric transformations on graphical objects.
- extract scene with different clipping methods and its transformation to graphics display device.
- learn projections techniques and basics of animation.

### Unit 1: Introduction to Computer Graphics

Definition, Application Areas of Computer Graphics (CAD, CAM, Education and Training, Entertainment, Image Processing, Computer Art etc.), Graphical User Interfaces (GUI)

### Unit 2: Graphics Systems

Cathode Ray Tubes, Random Scan Displays, Raster Scan Displays, Color CRT Monitors, FlatPanel Displays (Plasma-Panels, Liquid Crystal Displays(LCD), Electroluminescent displays), Graphics Software (GKS, PHIGS)

### Unit 3: Line Drawing Algorithms

Points and Lines, Line Drawing Algorithms (DDA Algorithm, Bresenham's Line Algorithm)

### Unit 4: Circle and Ellipse Drawing Algorithms

Circle drawing algorithms, Ellipse Drawing algorithms

### Unit 5: Filled Area Algorithms

Filling (Scan-Line Polygon filling, Inside outside tests, Boundary-fill and Flood-fill algorithm)

### Unit 6: Transformations

Basic 2-D Transformations (Rotation, Reflection, shearing, scaling), Homogeneous Coordinate Representation, Translation, 3-D transformations

### Unit 7: 2-D Viewing

2-D Viewing: The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions

### Unit 8: Clipping Algorithms

Line and polygon clipping algorithms (Cohen-Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm)

### Unit 9: 3-D Object Representation

3-D Object representation: Polygon surfaces, quadric surfaces, spline representation, Basic illumination models, polygon rendering methods

### Unit 10: Bezier and B-spline Curves

Hermite curve, Bezier curve and B-spline curves, Bezier and B-spline surfaces

### Unit 11: Projections

Projection Classification, Parallel projections, Perspective projections (One point, Two point).

### Unit 12: 3-D Geometric Transformations

Translation, rotation, scaling, reflection and shear transformations, composite transformations

### Unit 13: 3-D Viewing

Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping

### Unit 14: Basics in Animation

Animation Basic, Computer Animation, Principles of Animation, Types of Animation, Animation Software (Maya, etc) and Hardware, Common Terms in Animation

### Unit 15: Animation Designing

Design of Animation sequence, Morphing, Application of Computer Animation, Future of Animation

### References:

1. Donald Hearn and M Pauline Baker, Computer Graphics, PHI
2. James D. Foley, Andries Van dam, Steven K. Feiner & John F. Hughes, Computer Graphics - Principles and Practices, Pearson Education.
3. David F. Rogers, Procedural Elements for Computer Graphics, Tata-McGraw Hill

## SEMESTER VI: SOFTWARE ENGINEERING

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 30, Summative: 70)</b>

### Course Objective:

- To provide the concept of various software development methodologies for identifying suitable models for software development.
- To provide the understanding of various theories, models, and techniques which are the basis for software lifecycle.
- To provide learners the knowledge of software project management.
- To comprehend the use of different testing strategies in software development.

### Course Outcome:

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

- learn the principal steps for design, development and implementation of a software.
- identify and apply appropriate software architectures and patterns to carry out design of a system and be able to compare alternative choices.
- learn to select appropriate process models for software development.
- describe staffing, scheduling and about requirement analysis and specification.
- have experience and/or awareness of testing problems.

### Unit 1: Introduction to Software Engineering

Software Engineering, Need of Software Engineering, Software Characteristics, Software Development Projects: Programs versus Products, Quality and Productivity of software products.

### Unit 2: System Development Methodologies

System Development Phases, Software Life Cycle Model, Software Development Models: Classical and Iterative Waterfall Model, Prototyping Model, Evolutionary Model, Spiral Model, Comparison of different Models.

### Unit 3: Software Project Management

Software Project Manager, Project Planning, Project Estimation Techniques, Empirical method of Estimation, Heuristic method of Estimation- COCOMO Model, Analytical method of Estimation, Risk Management, Configuration Management.

### Unit 4: Staffing and Scheduling

Staffing, Scheduling, Work Breakdown Structure, Activity Networks, Critical Path, Gantt Charts, PERT Charts

### Unit 5: Role of System Analyst

System Analyst, Knowledge and Qualities of System Analyst, Role of a System Analyst.

**Unit 6: Requirement Analysis and Specification**

Fact Gathering, Requirement Analysis, Feasibility Study and Types, User Transaction Requirement, User Design Requirements, System Requirement Specification (SRS), Functional Requirements of SRS, Organization of SRS.

**Unit 7: Software Design**

Modules Concepts and Types of Modules, Structured Chart, Qualities of Good Design: Coupling, Types of Coupling, Cohesion, Types of Cohesion, Approaches to Software Design, Function-Oriented Design, Object-Oriented Design

**Unit 8: Function-Oriented Software Design**

Entity-Relationship Diagrams, Decision Tree and Decision Table, Data Flow Diagrams (DFD), Data Dictionary: Elements of DD, Advantage of DD; Pseudo code, Input And Output Design.

**Unit 9: Object-Oriented Software Design**

Unified Modeling Language (UML), UML Diagrams, Use Case Model, Class Diagrams, Activity Diagrams.

**Unit 10: User Interface Design**

Characteristics of a Good User Interface, Basic Concept: User Guidance and Online Help, Modeless Interface, Graphical User Interface (GUI) vs. Text-based User Interface, Types of User Interfaces, Fundamentals of Component-based GUI Development, A User Interface Design Methodology.

**Unit 11: Software Coding**

Coding, Coding Standards and Guidelines, Code Review: Code Walkthrough, Code inspection, Clean room testing, Software Documentation, Internal Documentation, External Documentation.

**Unit 12: Software Testing**

Software Testing, Testing Strategies, Types of Testing: Black-Box Testing, White-Box Testing, Stress Testing, Storage Testing, Performance Testing, Unit Testing, Integration Testing.

**Unit 13: Software Reliability and Quality Management**

Software Reliability, Hardware versus Software Reliability, Reliability Matrics, Reliability Growth Modelling, Statistical Testing, Software Quality, Software Quality Management System, Evolution of Quality Systems, Product Metrics versus Process Metrics

**Unit 14: Software Maintenance**

Software Maintenance, Types of Software Maintenance, Software Reverse Engineering, Maintenance Cost

**Unit 15: Case Tools**

Introduction to CASE, Building Blocks of CASE, Integrated CASE Environment.

**References:**

1. Mall, R. (2018). Fundamentals of software engineering. PHI Learning Pvt. Ltd.
2. Jalote, P., An Integrated approach to software engineering, Narosa Publishing House.
3. Pressman, R. S., Software Engineering: A practical Approach; McGraw-Hill.
4. Humphery, W. S., Managing software Procedures, Addison-Wesley

**SEMESTER VI: WEB TECHNOLOGY**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

**Course Objective:**

- To provide the knowledge of important components of the Internet and Web.
- To get introduced to HTML, DHTML, XML, Cascading Style Sheets, and basics of AJAX.
- To get introduced to JavaScript to build dynamic web pages.
- To get introduced to server-side programming PHP.

## **Course Outcome:**

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

- get introduced to core parts of a web.
- create web pages using HTML, DHTML, and Cascading Style Sheets.
- learn various HTML tags, table and list creation tags, HTML frames and forms.
- build dynamic web pages using the client-side programming JavaScript.
- create XML documents.
- learn the basics of AJAX
- build interactive web pages using PHP

### **Unit 1: Basics of Internet and Web**

The basics of Internet, World Wide Web, Web page, Home page, Web site, Static, Dynamic and Active web page, Overview of Protocols – Simple Mail Transfer Protocol, Gopher, Telnet, Emails, FTP, Hyper Text Transfer Protocol, Client server computing concepts, MIME, Web Surfing

### **Unit 2: Web Client**

Web Browser, Browsers e.g., Netscape navigator, Internet Explorer, Mozilla Firefox, Client-Side Scripting Languages- VB Script and Java Script, Active X control and Plug-ins

### **Unit 3: Web Sever**

Web Server Architecture, Image maps, CGI, API web database connectivity-DBC, ODBC, Proxy Server

### **Unit 4: Introduction to HTML**

Introduction to HTML, Essential Tags, Tags and Attributes, Text Styles and Text Arrangements, Text, Effects, Exposure to Various Tags (DIV, MARQUEE, NOBR, DFN, HR, LISTING, Comment, IMG), Color and Background of Web Pages, Attributes of Image Tag, Hypertext, Hyperlink and Hypermedia, Links, Anchors and URLs, Links to External Documents.

### **Unit 5: HTML Lists and Tables**

Lists, Unordered Lists, Ordered Lists, Definition Lists, Tables, TABLE, TR and TD Tags, Cell Spacing and Cell Padding, Colspan and Rowspan

### **Unit 6: HTML Frames and Form**

Frame, Frameset, FRAME Tag, NOFRAMES Tag, Forms, FORM and INPUT Tag, Text Box, Radio Button, Checkbox, SELECT Tag and Pull Down Lists, Hidden, Submit and Reset

### **Unit 7: Cascading Style Sheet**

Cascading Style Sheets (CSS), Inline Style, Embedded Style, External Style Sheet, Imported Style Sheet, Creating a CSS file, Using CSS in a web page

### **Unit 8: Dynamic HTML**

Dynamic HTML, Document Object Model, Features of DHTML, CSSP (Cascading Style Sheet Positioning) and JSSS (JavaScript assisted Style Sheet), Layers of Netscape, The ID Attribute, DHTML Events.

### **Unit 9: Introduction to JavaScript**

Introduction, Language Elements: Identifiers, Expressions, Operators, Statement Functions

### **Unit 10: Objects in JavaScript**

Objects: Window, Document, Form Objects, TextBox, TextArea, Button, Radio Button, CheckBox, Select Objects, Date, Match, String Objects; Regular Expression, Arrays

### **Unit 11: AJAX Basics**

Introduction, AJAX, XML http Request Object, AJAX Request, AJAX Response, AJAX Events

### **Unit 12: XML**

XML - Declaration, Root Element, Child Elements, Element Attributes, Entity References, Comments.

### **Unit 13: Introduction to PHP**

Server Side Programming , Introduction to PHP, Basic Programming Concepts of PHP: Variables, Data-types, Constants, Scope of Variables, Type of Variables, Type Casting, Operators, Operators Precedence, References.

### **Unit 14: Control statements in PHP**

Control Structures: Branching, If statement, Switch statement; Looping: for Loop, while Loop, do while Loop, for each Loop; Arrays

### **Unit 15: Functions in PHP**



Functions: User Defined Functions, Built-in Function, Functions for Variables; Script Controlling Functions, Array Functions, Date and Time Functions, Mathematical Functions, String Functions, PHP Server Variables; Working with form, Uploading files to Web Server using PHP.

**References:**

1. Steven Holzner,HTML Black Book, Dremtech press.
2. Jeffrey C. Jackson ,Web Technologies: A Computer Science Perspective, Prentice Hall, 2006.
3. Craif Knuckles, David Yuen, Web Applications: Concepts and Real-World Design, Wiley India Pvt Ltd, 2006.

**SEMESTER VI: INTRODUCTION TO PYTHON PROGRAMMING**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

**Course Objective:**

- To get introduced to Python programming.
- To provide the concept of various flow control of the Python programming language such as conditionals, looping.
- To provide the concept of functions and its implementation in Python.
- To provide the ability to design an object – oriented programs with Python.
- To provide the knowledge of file processing in Python

**Course Outcome:**

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

- learn Python installation, Anaconda installation
- work with Google Colab, Jupyter Notebook, and Spyder.
- learn Python basics like input/output statements, strings, decision control and loop control statements.
- define and demonstrate the use of built-in data structures lists, tuples, and dictionary.
- use functions and modules in Python programming.
- design object-oriented programs and to handle files in Python.

**UNIT 1: Installation of Python**

Python Installation, Working with Google Colab, Anaconda Installation, Working with Jupyter Notebook, Spyder

**UNIT 2: Introduction to Python**

Python, Python Interpreter, Using Python as calculator, Python shell, Indentation.

**UNIT 3: Python Basics**

Atoms, Identifiers and keywords, Literals, Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment operator, Bit wise operator, Identity and Membership operator.

**UNIT 4: Strings**

Strings, String properties, Built-in functions, String Methods, String Conversions, String Comparisons

**UNIT 5: Decision Control Statements**

Input and Output Statements, Control statements, Conditional Statements- if...else, Nested if statements, Ternary Operators.

**UNIT 6: Loop Control Statements**

Looping- while Loop, for Loop, Loop Control, break, continue and pass.

**UNIT 7: Lists**

Lists, Accessing List elements, looping in Lists, Basic List operations, Using built-in functions on Lists, List methods, Sorting and reversing.

**UNIT 8: Tuples**

Tuples, accessing tuple elements, looping in tuples, basic tuple operations, using built-in functions on Tuples, Tuple methods, Tuple varieties

#### **UNIT 9: Sets**

Definition of set, accessing set elements, looping in sets, basic set operations, Sets, Sets Methods and Functions

#### **UNIT 10: Dictionaries**

Dictionary, definition of dictionaries, accessing dictionary elements, looping in dictionaries, basic dictionary operations.

#### **UNIT 11: Functions**

Functions: Definition, call, positional and keyword parameter. Default parameters, variable number of arguments

#### **UNIT 12: Modules**

Modules: import mechanisms, Functional programming: map, filter, reduce, max, min. lambda function.

#### **UNIT 13: Object Oriented Programming**

Object Oriented Programming, Definition of Classes and Objects, Python Inheritance, Different types of Inheritance with examples.

#### **UNIT 14: File Processing**

File Processing: reading and writing files, manipulating file pointers using seek.

#### **References:**

1. Budd, T. (2010). *Exploring Python*. McGraw Hill Higher Education.
2. [The Python Tutorial — Python 3.11.2 documentation](#)
3. Wentworth, P., Elkner, J., Downey, A. B., & Meyer, C. (2015). *How to think like a computer scientist: Learning with Python 3*.

### **SEMESTER VII: FULL STACK DEVELOPMENT**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

#### **Course Objective:**

- Learners will learn the basics of web development using HTML, CSS, and JavaScript. They will gain a deep understanding of these technologies, including how to structure web pages, style content, and add interactivity with JavaScript.
- Learners will gain a deep understanding of ReactJS, including how to use JSX, create components, manage state, and handle user input. They will learn how to build robust, dynamic web applications that can scale to meet the needs of users.
- Learners will learn how to use NodeJS to create fast, efficient web servers and APIs. They will also gain a deep understanding of MongoDB and how to use it to store and retrieve data. They will be able to build full-stack web applications that can handle large amounts of data.

#### **Course Outcome:**

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

- develop dynamic and interactive web applications using HTML, CSS, JavaScript, ReactJS, NodeJS, MongoDB, Python, and PHP frameworks.

- create and manage databases using MongoDB, including migrating data into MongoDB and integrating it with NodeJS, Python and PHP
- understand and apply key concepts such as templating, components, state, props, lifecycle, routing, and error handling in ReactJS, as well as features such as Redux, Redux Saga, and service side rendering.

#### **Unit 1: HTML (Part 1)**

Introduction to HTML, Browsers and HTML, Editor's Offline and Online, Tags, Attribute and Elements, Doctype Element, Comments, Headings, Paragraphs, and Formatting Text, different versions of HTML.

#### **Unit 2: HTML (Part 2)**

Inserting Images, Hyperlinks, creation of Tables, Table attributes, Creation of Lists, Types of Lists.

#### **Unit 3: CSS**

Introduction CSS, Applying CSS to HTML, Selectors, Properties and Values, CSS Colors and Backgrounds, CSS Box Model, CSS Margins, Padding, and Borders, CSS Text and Font Properties, CSS General Topics.

#### **Unit 4: Programming with Javascript (Part 1)**

Introduction to JavaScript, Applying JavaScript (internal and external), Understanding JS Syntax, Introduction to Document and Window Object, Variables and Operators, Data Types and Num Type Conversion, Math and String Manipulation, Objects and Arrays, Date and Time.

#### **Unit 5: Programming with Javascript (Part 2)**

Conditional Statements, Switch Case, Looping in JS, Functions.

#### **Unit 6: ReactJS Development (Part 1)**

Introduction to ReactJS, Templating using JSX, Components, State and Props.

#### **Unit 7: ReactJS Development (Part 2)**

Lifecycle of Components, Rendering List and Portals, Error Handling, Routers.

#### **Unit 8: ReactJS Development (Part 3)**

Redux and Redux Saga, Immutable.js, Service Side Rendering, Unit Testing, Webpack.

#### **Unit 9: NodeJS Development (Part 1)**

Node js Overview, Node js - Basics and Setup, Node js Console, Node js, Command Utilities.

#### **Unit 10: NodeJS Development (Part 2)**

Node js Modules, Node js Concepts.

#### **Unit 11: NodeJS Development (Part 3)**

Node js Events, Node js with Express js, Node js Database Access.

#### **Unit 12: MongoDB (Part 1)**

QL and NoSql Concepts, Create and Manage MongoDB, Migration of Data into MongoDB.

#### **Unit 13: MongoDB (Part 2)**

MongoDB with NodeJS, Services Offered by MongoDB.

#### **Unit 14: Python**

Python Installation & Configuration, developing a Python Application, Connect MongoDB with Python.

#### **Unit 15: PHP**

PHP installation and Configuration, Developing PHP application, connect MongoDB with PHP, PHP and MongoDB operations with examples.

#### **References:**

1. "HTML and CSS: Design and Build Websites" by Jon Duckett
2. "Learning React: A Hands-On Guide to Building Web Applications Using React and Redux" by Kirupa Chinnathambi.
3. "Node.js, MongoDB, and AngularJS Web Development" by Brad Dayley.

## SEMESTER VII: INTRODUCTION TO IMAGE PROCESSING

Course Type:	DSC
Number of Credits:	4
Total Marks:	100 (Formative: 20, Practical: 30, Summative: 50)

### Course Objective:

- To provide image fundamentals and mathematical transforms necessary for image processing.
- To provide the knowledge of image sampling, image transformation, and image enhancement techniques.
- To provide the concept of filtering and segmentation.
- To provide the knowledge of image restoration procedures.

### Course Outcome:

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

- learn the fundamental concepts of image processing.
- evaluate the techniques for image enhancement and image restoration.
- categorize and interpret various compression techniques.
- interpret image segmentation and representation techniques.

### Unit 1: Fundamentals of Image

Definition of image, Analog image, digital image, Advantages & disadvantages of digital image, digital image representation, pixel, neighbours of pixel

### Unit 2: Image Sampling

Introduction, Theory of 2D sampling, violation of sampling criterion, quantization, Resolution, Anatomy of the human visual system

### Unit 3: Digital Image

Classification of digital image: Raster image, vector image, Image types: Binary image, grayscale image, color image, multispectral image, hyper spectral image, Elements of image processing system: Image sensor, acquisition, CCD sensor, CMOS image sensor comparison of CCD and CMOS sensors, Digital camera, Image file format, GIF file format, JPEG, PNG, TIFF, PSD, EPS, BMP, Application of digital image processing.

### Unit 4: Image Transform

Introduction, need for transform, Image transform, classification of image transform, Fourier transform, 2D discrete Fourier transform, Properties of 2D DFT: Separable, spatial shift property, periodicity, convolution, correlation properties, scaling property, rotation property

### Unit 5: Image Enhancement in Spatial Domain

Introduction, Image enhancement in spatial domain: point operation, mask operation, Types of point operation: Brightness modification, contrast adjustment

### Unit 6: Image Histogram

Introduction, histogram equalization, procedure to perform histogram equalization

### Unit 7: Filtering

Introduction, Spatial filtering, linear filtering, low pass filter, limitations of low pass filter, weighted average filter, Bartlett filter, Gaussian filter, Median filter, High pass filtering

### Unit 8: Image Enhancement in Frequency Domain

Introduction, Low pass filtering in frequency domain, High pass filter in frequency domain, Butterworth high pass filter, Homomorphic filter

### Unit 9: Image Restoration

Introduction, Image degradation, Types of image Blur, Classification of Image Restoration technique, Image restoration model, Linear Image Restoration techniques, Pseudo Inverse filter, Wiener filter, Non-Linear Image Restoration techniques: Iterative method, Maximum likelihood Method, Stochastic image restoration technique.

**Unit 10: Image Noise**

Introduction, classification of noise image: adaptive noise, Multiplicative noise, impulse noise, Median filtering, trimmed average filter

**Unit 11: Image Segmentation**

Introduction, Classification of Image segmentation technique, Local and global segmentation, region approach to image segmentation, clustering technique, Image segmentation based on thresholding, Limitation, edge detection techniques, edge linking, Watershed transformation

**Unit 12: Image Compression**

Introduction, Need for Image compression, Redundancy in images, Image compression scheme Huffman coding, Arithmetic coding, transform base coding, Vector quantization

**Unit 13: Binary Image Processing**

Introduction, Binarisation, Mathematical morphology, structuring element, Morphological image processing, Basic set theory, Logical operations, Standard binary morphological operations: Dialation, Erosion, Opening, Closing, Properties of Morphological operations.

**Unit 14: Color Image Processing**

Introduction, Color formation, Human perception of color, Color model: RGB model, CMY color model, HIS color model, YIQ color model, Histogram equalization of color image, Color Image filtering

**Unit 15: Wavelet-based Image Processing**

Introduction, Wavelet, Wavelet transform, 2D continuous wavelet transform, Discrete wavelet transform, Haar Wavelet, Filter bank, Embedded Image coding.

**References:**

1. Digital Image Processing, Rafael C. Gonzalez and Richard E. Woods, 4th Edition, Prentice Hall.
2. Anil K. Jain, Fundamentals of Digital Image Processing, Prentice Hall.
3. Stan Birchfield, Image Processing and Analysis, Cengage Learning.

**SEMESTER VII: DATA MINING AND DATA WAREHOUSING**

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 30, Summative: 70)</b>

**Course Objective:**

- To introduce learners to the basic concepts and techniques of Data Mining and Data Warehousing.
- To introduce basic principles, concepts and applications of data warehousing.
- To introduce the concept of data preprocessing.
- To provide the knowledge of data warehouses architecture.
- To introduce a wide range of clustering, estimation, prediction, and classification algorithms.

**Course Outcome:**

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

- understand the functionality of the various data mining and data warehousing components.
- remove redundancy and incomplete data from the dataset using data preprocessing methods.
- learn about interesting patterns from large amounts of data to analyze for predictions and classification.
- learn about clustering and various clustering algorithms.
- understand warehousing architectures and tools for systematically organizing large databases.
- learn about OLAP and OLTP.
- characterize the kinds of patterns that can be discovered by association rule mining.

**Unit 1: Introduction to Data Mining**

Data mining, various types of Data, Data Mining Functionalities, Classification of Data mining systems, Data mining Task Primitives, Integration of Data Mining System, Major issues in Data Mining

**Unit 2: Introduction to Data Warehousing**

Data Warehouse and DBMS, Need for data warehousing, Operational & Informational Data Stores, Data Warehouse Characteristics, Building a Data Warehouse, Design/Technical/Implementation Considerations, Data Warehouse role & Structure, The cost of warehousing data

**Unit 3: Introduction to OLAP**

Introduction to OLAP & OLTP, Difference between OLAP & OLTP, OLAP Operations

**Unit 4: Data Preprocessing**

Data preprocessing, Data Summarization, Data Cleaning, Data Transformation, Data reduction, Concept Hierarchy, Structure

**Unit 5: Multidimensional Data**

Multidimensional Data Model, Schemas for Multidimensional Data (Star Schema, Snowflake Schema, Fact Constellation)

**Unit 6: Data Warehouse Architecture**

Data Warehouse Architecture, Data Warehouse Design, OLAP Three -tier Architecture, Indexing & Querying in OLAP, OLAM, Implementation from Data Warehouse to Data mining

**Unit 7: Data Mining Knowledge Representation**

Task relevant data, Background knowledge, Interestingness measures, Representing input data and output knowledge, Visualization techniques

**Unit 8: Attribute-oriented Analysis**

Attribute generalization, Attribute relevance, Class comparison, Statistical measures

**Unit 9: Association Rule Mining**

Association Rule Mining, Market Basket Analysis, Apriori Algorithm, Mining Multilevel Association Rules, From Association Mining to Correlation Analysis

**Unit 10: Classification**

Introduction to Classification, Classification by Decision Tree, Attribute Selection Measure, Covering rules

**Unit 11: Prediction**

Introduction to Prediction techniques, Statistical (Bayesian) classification, Bayesian networks, Instance-based methods (nearest neighbor)

**Unit 12: Evaluation**

Training and Testing, Evaluating accuracy of a Classifier, Cross-Validation, Combining multiple models (Bootstrap, Boosting, Bagging)

**Unit 13: Clustering**

Introduction to Clustering, Classification of Various Clustering Algorithms, Partitioning methods- k-means, Density based DB-SCAN, Hierarchical methods- agglomerative and divisive clustering

**Unit 14: Introduction to Web Mining**

Web Mining introduction, web content mining, web structure mining, web usage mining, text mining, unstructured text, episode rule discovery for text

**Unit 15: Introduction to Spatial and Temporal Data Mining**

Temporal Data Mining, Temporal Association Rules, Sequence Mining, Spatial Mining, Spatial Mining tasks, spatial clustering.

**References:**

1. Ville, Decision Trees for Business Intelligence and Data Mining: Using SAS Enterprise Miner, SAS, 2006.
2. Pang-Ning Tan, Introduction to Data Mining, Addison Wesley, 2006.
3. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufman, 2001.
4. Tom Soukup, Davidson, Visual Data Mining: Techniques and Tools for Data Visualization and Mining, Wiley, 2002.
5. Alex Berson, Stephen J. Smith, Data Warehousing, Data Mining, and OLAP, MGH, 1998.

## SEMESTER VIII: OPEN SOURCE SOFTWARE

Course Type:	<b>DSC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

### Course Objective:

- To familiarize learners to the free open-source software environment and procedure for installing open-source packages.
- To provide the knowledge of using open-source packages LaTeX, Scilab, and Python with their applications.
- To provide a clear understanding of the basics of technical writing which will enable the learners to communicate their ideas effectively in the form of technical reports, journal papers etc., by using the technical writing tool LaTeX.
- To provide the knowledge of the Scilab environment and programming language.
- To get introduced to the basics of Python programming.

### Course Outcome:

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

- understand the installation of various open-source packages like LaTeX, Scilab, and Python.
- formatting documents containing sectioning, tables, equations and figures.
- typesetting articles, technical reports, slide presentations, books etc.
- learn the Scilab environment and programming language.
- learn about the basics of Python commands and programming.

### Unit 1: Installation to LaTeX

Installation of LaTeX, Understanding Latex compilation

### Unit 2: Introduction of LaTeX

Basic Syntax, Writing equations, Matrix, Tables

### Unit 3: Page Layout – I

Page Layout – Titles, Abstract Chapters, Sections, References, Equation references, citation, Table of contents

### Unit 4: Page Layout – II

List making environments, Generating new commands, Figure handling, table & figure numbering, List of figures, List of tables, Generating index

### Unit 5: Packages

Packages: Geometry, Hyperref, amsmath, amssymb, algorithms, algorithmic graphic, color, tilez listing

### Unit 6: Classes

Classes: article, book, report, beamer, slides, letter

### Unit 7: Applications of LaTeX

Applications: Writing Resume, Writing question paper, Writing articles/ research papers, Presentation using beamer, inserting graphics, drawing graphics, putting equations

### Unit 8: Installation of Scilab

Installation of Scilab (both windows & Linux)

### Unit 9: Introduction to Scilab

Introduction to scilab, Basic syntax, Mathematical Operators, Predefined constants, Handling .sci files, Installation of additional packages e.g. optimization

### Unit 10: Functions in Scilab

Built in functions, Complex numbers, Polynomials, Vectors, Matrix operations (functions like inv(), spec(), zeros(), ones(), eye(), rand()). Handling these data structures using built in functions.

**Unit 11: Conditional Statements in Scilab**

Loops (for & while), Conditional statements

**Unit 12: Graphics in Scilab**

Graphics handling: 2D, 3D, Generating .jpg files, Function plotting, Data plotting, GUI in Scilab, Plotting 2D graphs

**Unit 13: Applications of Scilab**

Applications: Numerical Linear Algebra (Solving linear equations, Eigen values etc.), Solving Ordinary Differential Equations, Numerical Analysis – iterative methods, Comparison with C/C++/Matlab

**Unit 14: Introduction to Python**

Introduction to Python, Procedure to install Python, How to open Python console, Basic Python commands, Python programming

**References:**

1. F. Mittelbach, M. Goossens, The LaTeX Companion, 2nd ed. Addison- Wesley (2004), ISBN 0-201-36299-6.
2. L. Lamport: LATEX, A Document Preparation System, User's Guide and Reference Manual, 2nd ed., Addison-Wesley (1994), ISBN 0-201-52983-1
3. Knuth, D. E., & Bibby, D. (1984). The Tex book (Vol. 3). Reading: Addison-Wesley.
4. Verma, A. K., (2018). SCILAB: A beginner's Approach.
5. Nagar, S., (2018). Introduction to Scilab: For Engineers and Scientists.
6. Barry, P. (2016). Head First Python: A Brain-Friendly Guide. " O'Reilly Media, Inc."
7. Matthes, E. (2015). Python crash course: a hands-on, project-based introduction to programming. No Starch Press.

### SEMESTER VIII: FUNDAMENTALS OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Course Type:	<b>Core</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 30, Summative: 70)</b>

**Course Objective:**

- To provide learners an introduction to Artificial Intelligence.
- To provide the fundamental concept of Machine Learning.
- To provide the understanding of supervised and unsupervised learning

**Course Outcome:**

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

- learn about the basic concepts of Artificial Intelligence.
- explain the basics of problem solving, and learning methods of AI.
- learn the techniques on how to make learning by a model, how it can be evaluated, what are all different algorithms to construct a learning model.
- get introduced to the field of Machine Learning, in particular focusing on the concepts of supervised and unsupervised learning.



## **BLOCK 1: ARTIFICIAL INTELLIGENCE**

### **Unit 1: Introduction to Artificial Intelligence**

What is intelligence? Foundations of Artificial Intelligence (AI). History of AI, Definition of AI.

### **Unit 2: Problem Solving in AI**

Problem Solving- Formulating problems, problem types, states and operators, state space, search strategies.

### **Unit 3: Informed Search Strategies**

Best first search, A\* algorithm, heuristic functions.

### **Unit 4: Game playing**

Perfect decision game, imperfect decision game, evaluation function, alpha-beta pruning.

### **Unit 5: Reasoning**

Representation, Inference, Propositional Logic, predicate logic (first order logic), logical reasoning, forward chaining, backward chaining; AI languages and tools- Lisp, Prolog, CLIPS.

### **Unit 6: Planning**

Basic representation of plans, partial order planning, planning in the blocks world, hierarchical planning, conditional planning, representation of resource constraints, measures, temporal constraints.

### **Unit 7: Uncertainty**

Basic probability, Bayes rule, Belief networks, Default reasoning, Fuzzy sets and fuzzy logic; Decision making- Utility theory, utility functions, Decision theoretic expert systems.

### **Unit 8: Inductive Learning**

Decision trees, rule-based learning, reinforcement learning, genetic algorithms;

### **Unit 9: Communication**

Communication among agents, natural language processing.

## **BLOCK 2: MACHINE LEARNING**

### **Unit 10: Introduction to Machine Learning**

Definition of Learning systems; Goals and Applications of Machine learning, Aspects of developing a learning system, Training data, Problems, Data and Tools, Supervised Learning vs. Unsupervised Learning.

### **Unit 11: Classification**

Probability and classification, Decision tree, Representing concepts as decision trees, Selection of the best splitting attributes.

### **Unit 12: Linear Regression**

Simple regression Analysis, Multiple Linear Regression Model

### **Unit 13: Support Vector Machine**

Support vector machines and large-margin classifiers, vector quantization.

### **Unit 14: Artificial Neural Networks**

Neurons and biological motivation, Linear threshold Units, Perceptron: representational limitation and gradient decent training. Multilayer networks and backpropagation. Hidden layers constructing intermediate, distributed representations.

### **Unit 15: Clustering and Unsupervised Learning**

Learning from unclassified data, Mixture densities, k-means clustering, partition-based clustering, hierarchical clustering.

## **References:**

1. E. Rich and K. Knight, "Artificial intelligence", Tata Mc Graw Hill Publications
2. N.J. Nilsson, "Principles of AI", Narosa Publ. House Publications
3. Alex Smola and S.V.N. Vishwanathan," Introduction to Machine Learning", Cambridge University Press 2008.
4. Andreas C. Müller and Sarah Guido," Introduction to Machine Learning with Python: A Guide for Data Scientists", O'Reilly Media, Inc,2016.
5. S. Russel and P. Norvig, "Artificial Intelligence: A Modern Approach", Third Edition, Prentice Hall, 2009.
6. C. M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2007.

# Syllabus of Minor Courses

## SEMESTER I: COMPUTER FUNDAMENTALS

Course Type:	Minor
Number of Credits:	4
Total Marks:	100 (Formative: 30, Summative: 70)

### Course Objectives:

- To provide learners the knowledge of computers, its different generations and applications
- To provide learners the knowledge of different components of computers, memory and operating system
- To familiarize students in using algorithms, flowcharts and pseudocodes to solve problems
- To provide knowledge of cybercrimes, computer security threats and protection mechanisms

### Course Outcome:

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

- describe components of computers, types of computer memory and different operating systems
- describe the different types of buses and instruction formats
- use algorithms, flowchart and pseudocode to solve problems
- describe computer security issues and explain different types of cyber crimes

### Unit 1: Introduction to Computer

Computer– a Definition; Characteristics of Computer; Generations of Computers; Structural Evolution of Computers; File Structure of Computers; Flynn’s Classification of Computer Architecture; Applications of Computer: Advantages of Computer

### Unit 2: Basic Features of Computer Classification

Computer Classification based on Operating Principles: Analog Computers, Digital Computers, Hybrid Computers; Computer Classification based on Applications: General Purpose Computers, Special Purpose Computers; Computer Classification based on Size and Capability: Micro Computers, Mini Computers, Mainframe Computers, Super Computers

### Unit 3: Basic Components of Computer

Concept of Bit and Byte; Block Diagram of Computer, CPU, ALU, CU; Computer Memory: Primary and Secondary; Input/Output Devices; Buses : Address, Data, Control.

### Unit 4: Computer Memory and Storage

Memory Representations; Memory Hierarchy; CPU Registers; Primary Memory: Random Access Memory, Read Only Memory; Secondary Memory; Cache Memory; Virtual Memory

### Unit 5: Buses

Bus Structure- Address, Data and Control Bus, Different types of Buses–ISA, PCI, EISA, VESA, MCA, CA

### Unit 6: Introduction to System Software

Compiler, Assembler, Linker, Loader and Editor

### Unit 7: Instruction Types

Different Types of Instruction format (8 bit, 16 bit), Bit manipulation, Control, Data Transfer Instruction

### Unit 8: Basics of Operating Systems

Functions of OS, Types of OS, DOS, LINUX (Single User, Multiuser), UNIX, ZENIX, WINDOWS, Multiprocessing OS, Time sharing OS

**Unit 9: Computer Languages**

Concepts of Machine level, Assembly level and High-level language; Basic difference between Procedure Oriented Language and Object-Oriented Language;

**Unit 10: Introduction to Algorithms and Flowchart**

Introduction to Algorithm, Flowchart, Pseudo code, Examples of algorithm and flowchart for simple problems.

**Unit 11: File Structure**

Data and Information, Records and its types, Files, Operations on Files

**Unit 12: Computer Security**

Computer Virus, Worm, SpyWare, Malware, Trojan Horse, Antivirus Software

**Unit 13: Basics of Cyber Crimes I**

Cyber Crimes against Individuals, Institution and State, Hacking, Digital Forgery, Identity Theft and Fraud.

**Unit 14: Basics of Cyber Crimes II**

Cyber Stalking/Harassment, Cyber terrorism, Cyber Defamation, Software Piracy, Different offences under IT Act, 2000

**References:**

1. P. K. Sinha & Priti Sinha , “Computer Fundamentals”, BPB Publications, 2007.
2. Dr. Anita Goel, “Computer Fundamentals”, Pearson Education, 2010.

**SEMESTER II: FUNDAMENTALS OF PC SOFTWARE**

Course Type:	<b>Minor</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 50)</b>

**Course Objectives:**

- To provide learners the knowledge of operating systems and using DOS operating system
- To provide learners the knowledge of using both Microsoft and Linux operating systems
- To familiarize students in usage of LibreOffice Writer and Calc.
- To familiarize students in usage of LibreOffice Impress and Base.

**Course Outcome:**

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

- use DOS operating system
- use Microsoft Windows and Linux operating system
- use LibreOffice Writer and Calc to create documents and worksheets
- use LibreOffice Impress and Base to create presentations and tables, queries and forms.

**Unit 1: Introduction to OS**

Definition and functions of an Operating System, Types of OS (Single User, Multi user, Single tasking, Multitasking, Real time, Network OS, Distributed OS)

**Unit 2: MS DOS Operating System**

Introduction to DOS, System files of DOS, concept of Booting, Files and Directory Structure, Concept of Paths, Internal and External commands, Batch File.

**Unit 3: MS Windows Operating System**

Introduction to Windows, Components of Windows, Customizing the Desktop, Files and Folders.

#### **Unit 4: LINUX Operating System**

Open Source and Free Software, Advantages and Disadvantages of Linux Operating System, Concept of Path, Basic Linux Commands, File Permission, Text Editor vi.

#### **Unit 5: Introduction to LibreOffice**

Overview of LibreOffice software suite and its features, Advantages of using LibreOffice as an alternative to proprietary software, Installation and configuration of LibreOffice on different platforms

#### **Unit 6: Getting started with Writer**

Launching Writer and creating a new document, Navigating the Writer interface and understanding the different components of the document window, Entering and formatting text, applying font styles, and working with paragraphs, Saving, closing, and opening existing documents in different formats

#### **Unit 7: Working with templates and styles in Writer**

Creating and using document templates for consistent formatting and layout, Creating and modifying paragraph and character styles for consistent text formatting, Managing styles in the Styles and Formatting dialog box

#### **Unit 8: Formatting documents in Writer**

Adding and formatting images, tables, and other objects in a document, Applying advanced formatting options, such as page borders, columns, and page backgrounds, Using Writer's built-in tools for spell-checking, grammar-checking, and language translation

#### **Unit 9: Introduction to Calc**

Overview of Calc and its features as a spreadsheet program, Launching Calc and creating a new spreadsheet, Navigating the Calc interface and understanding the different components of the worksheet window, Entering and formatting data, applying cell formatting, and working with formulas

#### **Unit 10: Working with functions and formulas in Calc**

Using built-in functions and formulas for common calculations in Calc, creating custom functions and formulas for complex calculations, Using conditional formatting to highlight data based on certain criteria

#### **Unit 11: Advanced formatting and charting in Calc**

Applying advanced formatting options, such as conditional formatting, data validation, and custom number formats, Creating and modifying charts to visualize data in different ways, Using Calc's built-in tools for filtering, sorting, and analyzing data

#### **Unit 12: Introduction to Impress**

Overview of Impress and its features as a presentation program, Launching Impress and creating a new presentation, Navigating the Impress interface and understanding the different components of the slide window, Adding and formatting slides, applying transitions, and customizing presentation settings

#### **Unit 13: Working with multimedia and animations in Impress**

Adding images, audio, and video to a presentation, Creating and modifying animations to enhance visual effects, Using Impress's built-in tools for recording and rehearsing presentations

#### **Unit 14: Introduction to Base**

Overview of Base and its features as a database management tool, Launching Base and creating a new database, Navigating the Base interface and understanding the different components of the database window, Creating tables, forms, queries, and reports using Base's built-in tools

#### **References:**

1. <https://documentation.libreoffice.org/assets/Uploads/Documentation/en/GS7.0/GS70-GettingStarted.pdf>

## **SEMESTER III: FUNDAMENTALS OF DATABASE MANAGEMENT SYSTEM**

Course Type:	<b>Minor</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 70)</b>

### **Course Objectives:**

- To provide learners knowledge of database architecture, data models and relational model
- To provide learners knowledge of functional dependencies and normalization
- To familiarize learners with SQL programming to create, alter and delete data and tables
- To provide learners knowledge of using GUI and connecting to a database

### **Course Outcome:**

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

- describe the database architecture, data models and relational model
- explain functional dependencies, normalization and the different types of keys
- use SQL to create, modify, delete tables and databases and write queries and subqueries
- use WAMP and phpMyAdmin

### **UNIT 1: File Structure**

Data and Information, Concept of Field, Key field; Records and its types, Fixed length records and Variable length records; Files, Operation on files, Primary file organization

### **UNIT 2: Database System**

Traditional file approach versus Database approach; Data independence, Database System, Database Architecture, The three level of architecture, Mapping, Database administrator, Database Management system, Types of DBMS, Merits and demerits of DBMS

### **UNIT 3: Data Models**

Conceptual model, Logical model, Physical model, Entity-Relationship (ER) model as a tool for conceptual design: Entities, Attributes and Relationships, Weak and Strong entities, Conversion of ER model into Relational schema, ER modelling symbols.

### **UNIT 4: The Relational Model**

Relational data model concepts, Integrity constraints: Entity integrity, Referential integrity, Domain Constraints.

### **UNIT 5: Keys**

Concept of keys, Composite key, Candidate key, Primary key, Alternate key, Foreign key, Defining Primary and Foreign keys in Database.

### **UNIT 6: Relational Database Design**

Database Design, Decomposition, Universal Relation, Functional Dependencies, Prime and Non-prime attributes.

### **UNIT 7: Normalization**

Normalization, First Normal form (1NF), Second Normal form (2NF), Third Normal form (3NF), Boyce-Codd Normal form (BCNF)

### **UNIT 8: Introduction to SQL**

Structured Query Language (SQL), Characteristics of SQL, Advantages of SQL, SQL Data types, Types of SQL commands, DDL, DML.

### **UNIT 9: Working with SQL Part I**

Creating a Database, Creating, Altering, and Deleting Tables, Inserting new data, Updating data (the WHERE Clause, The Logical AND and OR operator), Deleting Data.

### **UNIT 10: Working with SQL Part II**

The SELECT statement, Filtering results with the WHERE Clause; Logical Operators and Operator Precedence (Using logical operator NOT, BETWEEN, LIKE, IN); Ordering Results with ORDER BY (Joining columns-concatenation); Selecting data from more than one table.

### UNIT 11: Working with SQL Part III

Join, Types of Join, SQL commands: Select... From... Where... Group by ..... Having... Order by..., Tables, Queries, Sub Queries, Insert, Update and Delete operations, Constraints considers (NOT NULL, UNIQUE, Check Primary key, Foreign key)

### UNIT 12: Working with GUI

Introduction to WAMP, Installation of WAMP, Working with phpMyAdmin.

### UNIT 13: Database Recovery

Concept of database recovery, Backup of database, Types of database failure, Types of database recovery.

### UNIT 14: Database Security

Goals of database security, Discretionary Access Control, Mandatory Access control, Statistical Databases, Data Encryption.

### References:

1. Elmasri, R., Navathe, S. B., Elmasri, R., & Navathe, S. B. (2000). *Fundamentals of Database Systems*. Addison-Wesley.
2. Ramakrishnan, R., Gehrke, J., & Gehrke, J. (2003). *Database management systems* (Vol. 3). New York: McGraw-Hill.
3. Silberschatz, A., Korth, H. F., & Sudarshan, S. (2011). *Database system concepts*.
4. Schwartz, B., Zaitsev, P., & Tkachenko, V. (2012). *High performance MySQL: optimization, backups, and replication*. " O'Reilly Media, Inc."

## SEMESTER IV: OPEN SOURCE SOFTWARE

Course Type:	<b>Minor</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 70)</b>

### Course Objectives:

- To provide the knowledge of using open-source packages LaTeX, Scilab, and Python with their applications.
- To provide a clear understanding of the basics of technical writing which will enable the learners to communicate their ideas effectively in the form of technical reports, journal papers etc., by using the technical writing tool LaTeX.
- To provide the knowledge of the Scilab environment and programming language.
- To get introduced to basics Python.

### Course Outcome:

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

- understand the installation of various open-source packages like LaTeX, Scilab, and Python.
- format documents, typeset journal articles, reports, books, and slide presentations in Latex
- use Scilab environment to plot functions and data
- write code to solve simple problems in Python.

**Unit 1: Installation to LaTeX**

Installation of LaTeX, Understanding Latex compilation

**Unit 2: Introduction of LaTeX**

Basic Syntax, Writing equations, Matrix, Tables

**Unit 3: Page Layout – I**

Page Layout – Titles, Abstract Chapters, Sections, References, Equation references, Citation, Table of contents

**Unit 4: Page Layout – II**

List making environments, Generating new commands, Figure handling, table & figure numbering, List of figures, List of tables, Generating index

**Unit 5: Packages**

Packages: Geometry, Hyperref, amsmath, amssymb, algorithms, algorithmic graphic, color, tilez listing

**Unit 6: Classes**

Classes: article, book, report, beamer, slides, letter

**Unit 7: Applications of LaTeX**

Applications: Writing Resume, Writing question paper, Writing articles/ research papers, Presentation using beamer, inserting graphics, drawing graphics, putting equations

**Unit 8: Installation of Scilab**

Installation of Scilab (both windows & Linux)

**Unit 9: Introduction to Scilab**

Introduction to Scilab, Basic syntax, Mathematical Operators, Predefined constants, Handling .sci files, Installation of additional packages e.g. optimization

**Unit 10: Functions in Scilab**

Built in functions, Complex numbers, Polynomials, Vectors, Matrix operations (functions like inv(), spec(), zeros(), ones(), eye(), rand()). Handling these data structures using built in functions.

**Unit 11: Conditional Statements in Scilab**

Loops (for & while), Conditional statements

**Unit 12: Graphics in Scilab**

Graphics handling: 2D, 3D, Generating .jpg files, Function plotting, Data plotting, GUI in scilab, Plotting 2D graphs

**Unit 13: Applications of Scilab**

Applications: Numerical Linear Algebra (Solving linear equations, eigen values atc.) solving Ordinary Differential Equations, Numerical Analysis – iterative methods, Comparison with C/C++/Matlab

**Unit 14: Introduction to Python**

Introduction to Python, Procedure to install Python, How to open Python console, Basic Python commands, Python programming.

**References:**

1. Kopka, H., Daly, P. W., & Rahtz, S. P. Q. (2004). *Guide to LATEX* (Vol. 4). Boston, MA: Addison-Wesley.
2. Ramchandran, H., & Nair, A. S. (2012). *Scilab (a Free Software to Matlab)*. S. Chand Limited.
3. [The Python Tutorial — Python 3.11.2 documentation](#)

## SEMESTER V: FUNDAMENTALS OF WEB TECHNOLOGY

Course Type:	<b>Minor</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 70)</b>

### Course Objectives:

- To familiarize learners with Internet and its related topics
- To provide the knowledge of HTML programming
- To provide the knowledge of CSS
- To train a learner, so that he/she can create a simple website

### Course Outcome:

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

- describe basic concepts related to the Internet, WWW and various protocols associated with email, file transfer etc.
- describe various tags used in HTML for web page design
- explain the use of CSS with HTML
- design a website

### Unit 1: Basics of Internet and Web

Basics of Internet, World Wide Web, Web page, Home page, Web site, Static, Dynamic and Active web page, Overview of Protocols – Simple Mail Transfer Protocol, Gopher, Telnet, Emails, FTP, Hypertext Transfer Protocol, Client server computing concepts, MIME, Web Surfing

### Unit 2: Web Client

Web Browser, Browsers e.g., Google Chrome, Microsoft Edge etc, Client-Side Scripting Languages-VB Script and Java Script, Active X control and Plug-ins

### Unit 3: Web Server

Web Server Architecture, Image maps, CGI, API web database connectivity-DBC, ODBC, Proxy Server

### Unit 4: Introduction to HTML

HTML markup, History of HTML, Use of HTML, tags, elements, attributes, creating HTML text using Notepad, Standard HTML document structure format

### Unit 5: HTML Formatting-I

The browser as formatter, comment, Paragraph, line break, preformatting, heading, horizontal rules, meta tags, HTML styles, Working with other HTML formatting elements

### Unit 6: HTML Formatting-II

Working with colors, hyperlink, images, lists, tables

### Unit 7: HTML iframes

Iframe syntax, setting height and width, removing the border, target for a link, case studies

### Unit 8: HTML Class and ID

Class attribute, syntax for class, multiple classes, id attribute, difference between class and id, HTML Bookmarks with ID and Links

### Unit 9: HTML Layout and Responsive

HTML layout, HTML responsive, HTML emojis

### Unit 10: HTML Forms

HTML Forms, HTML Form Attributes, HTML Form Elements, HTML Input Types, HTML Input Attributes, HTML Input Form Attributes.

### Unit 11: HTML Media

HTML Media, Video, Audio, Plug-ins, YouTube

### Unit 12: Working with CSS-I

Definition, Use of CSS, Types of CSS with examples



### Unit 13: Working with CSS-II

Working with internal CSS, inline CSS, external CSS

### Unit 14: Case Studies

Creating a simple website using HTML and CSS

#### References:

1. Duckett, J., & Schlüter, J. (2011). *HTML and CSS*. Wiley.
2. Duckett, J. (2011). *Beginning html, xhtml, css, and javascript*. John Wiley & Sons.

## SEMESTER VI: DIGITAL MARKETING

Course Type:	Minor
Number of Credits:	4
Total Marks:	100 (Formative: 30, Summative: 70)

#### Course Objectives:

- To explain about different digital marketing strategy
- To identify how to create effective digital content that engages and informs customers, including the use of video, images, and written content
- To utilize digital advertising tools
- To identify latest trends and developments in digital marketing

#### Course Outcome:

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

- create and execute effective digital marketing campaigns
- explain data analytics and measurement
- analyze and interpret digital marketing data
- to effectively manage a digital marketing budget

#### Unit 1: Basics of Digital Marketing

Definition; Importance of Digital Marketing; working of digital marketing; advantages of digital marketing over traditional marketing; types of digital marketing

#### Unit 2: Keyword Research

Understanding keyword and its types; keyword research: tools for keyword research, localized keyword research; analysis of competitor website keyword; finalizing keyword for a project.

#### Unit 3: Website Planning

Importance of Domain name and domain extension; types of website: based on functionality and purpose; Planning of website design: content, plugin, engagement options; landing page and optimization.

#### Unit 4: Search Engine Optimization

Introduction to search engine and how it works; Concept of search engine optimization; Understanding the Search Engine Results Page; indexing; crawling

#### Unit 5: Social Media Marketing

Understanding social media and its marketing tools: paid marketing, organic marketing; Social Media optimization, Understanding marketing through Facebook, Instagram, YouTube, Google

#### Unit 6: Google mapping and Local Business

Listing of local in search engine; Image, video and Map setup in google places; visibility report of search engine and google reviews.

#### Unit 7: Google Analytics

Introduction and working of Google Analytics; Google analytics- account structure, insights, cookie

training; setting up analytics account; how to add google analytics in website

#### **Unit 8: WhatsApp Marketing**

Understanding WhatsApp Marketing; WhatsApp push notification; WhatsApp Business; tools and settings in WhatsApp business; Limitations of WhatsApp

#### **Unit 9: Facebook marketing**

Introduction to Facebook-insights, algorithm; Understanding Facebook ads- organic and paid, targeting option; Facebook page management

#### **Unit 10: Twitter Marketing**

Introduction to Twitter Marketing; Twitter marketing insights- dos and don'ts; Ad Content and targeting; fundamental of Influencer marketing; use of hashtags, tools

#### **Unit 11: Instagram Marketing**

Introduction to Instagram platform. Creation of Business page in Instagram; Marketing tools and Ads; Drive engagement and Lead generation; account switching

#### **Unit 12: YouTube Marketing**

Channel creation in YouTube; Uploading video and optimization; Video creation and story lineup; monetization of YouTube Channel; Case study

#### **Unit 13: E-commerce marketing**

E-commerce website and its advantages; marketing concept; Motivating shopping for online; affiliation marketing

#### **Unit 14: Social Media Automation**

Social Media automation- tools and process; Automatized report, analytics, best time to share; Rules related to automation

#### **Unit 15: Google Ad Sense and AdWords**

AdSense and process of earning through it; process of approval of AdSense; Placing Ads in website; AdWords and Ad placement; approval for AdWords; how to get Ads in Blog/YouTube

#### **References:**

1. Chaffey, D., & Ellis-Chadwick, F. (2019). *Digital marketing*. Pearson uk.
2. Ryan, D. (2016). *Understanding digital marketing: marketing strategies for engaging the digital generation*. Kogan Page Publishers.
3. Kingsnorth, S. (2022). *Digital marketing strategy: an integrated approach to online marketing*. Kogan Page Publishers.
4. Charlesworth, A. (2014). *Digital marketing: A practical approach*. Routledge.

## **SEMESTER VII: DIGITAL FLUENCY**

Course Type:	<b>Minor</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 70)</b>

#### **Course Objectives:**

- To familiarise learners to various eLearning and MOOCs Platforms and provide the working knowledge of various searching techniques and services provided by Google
- To give learners the concepts of various E-Payment methods
- To familiarise the learners with various video conferencing tools and proper handling of various social networking platforms
- To familiarize learners with concepts and applications of Artificial Intelligence, Machine Learning, Internet of Things, Cloud Computing, Big Data, e-Governance and Blockchain Technology.

## **Course Outcome:**

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

- understand the concept of eLearning and various eLearning platforms
- demonstrate how to create and work with email through Gmail, prepare and share documents through google docs, google sheets and manage and share files in google drive
- handle various E-payment method, e-Governance applications and social networking sites in a proper way
- understand the concepts and applications of Artificial Intelligence and Machine Learning, Internet of Things, Cloud Computing, Big Data and Blockchain technologies

### **Unit 1: Introduction to eLearning**

Definition, Asynchronous/Synchronous Learning, e-learning Platforms: Proprietary and open source, Learning Management System (LMS), MOOCs: definition, various MOOC platforms like SWAYAM, Coursera, Edx, Audacity, FutureLearn etc.

### **Unit 2: Searching Techniques**

Search Engines, Searching Techniques; Google, Bing, Advanced Search Options in Google, Translator: Google, Bing

### **Unit 3: Working with Google Services I**

Working with Gmail, Google Docs: creating and posting; Google Sheets: creating and editing; Google Drive: uploading and sharing of files and folders.

### **Unit 4: Working with Google Services II**

Google Forms: Creating and analysis of response; Google Voice Assistant; Google Classroom: Creating classroom, Taking class etc.

### **Unit 5: E-payment methods**

E-cash Payment System, Credit Payment system, E-Check, Concept of OTP and uses, Concept of QR code and uses, Overview of Smart Card; Types of Electronic Payment Systems: Credit Card, Debit Card, Smart Card, E Money, Electronic Fund Transfer (EFT), UPI, Google Pay, BHIM etc.

### **Unit 6: E-commerce**

Introduction, Types of E-commerce, Applications of E-commerce with examples.

### **Unit 7: Video Conferencing Tools**

Introduction, Working with various video conferencing tools like Google Meet, Zoom, WebX, Microsoft Teams, etc.

### **Unit 8: Working with Social Networks**

Introduction, Working with various Social Networking Sites like WhatsApp, Youtube, Telegram, Facebook, Twitter, Skype, Instagram etc.

### **Unit 9: Introduction to Artificial Intelligence and Machine Learning**

Definition of Artificial Intelligence (AI); Brief history of AI; Types/Categories of AI; Applications of AI: Healthcare, Agriculture, Education etc.; Machine Learning: definition, Supervised learning, Unsupervised learning; Introduction to Industry 4.0 and applications, Introduction to Education 4.0 and applications

### **Unit 10: Introduction to IoT**

Definition, basic concepts of IoT and its applications like Smart Home, Smart vehicles, IoT in Agriculture.

### **Unit 11: Introduction to Cloud Computing**

Cloud Computing: Definition; Types of Cloud: Public cloud; Private cloud; Hybrid cloud; Community cloud; Types of Cloud services: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS), Applications of Cloud Computing

### **Unit 12: Introduction to Big Data**

Big Data: Definition, Characteristics, Types of Big Data: Structured, Un-structured, Semi-structured; Apache Hadoop; Big Data Platforms, Applications of Big Data

### **Unit 13: E-Governance and Application**

E-Governance Models: (G2B,G2C,C2G,G2G), Challenges to E-Governance, Strategies and tactics for implementation of E-Governance, Applications of E-Governance

## Unit 14: Blockchain Technology

Introduction to Blockchain Technology and its Importance, Evolution of the Blockchain Technology, Elements of a Blockchain, Blockchain Applications

### References:

1. Gerald Knezek and Rhonda Christensen. *Digital Fluency: Skills Necessary for Learning in the Digital Age*
2. Kristin Fontichiaro. *Teaching Digital Fluency: How do we Prepare Students for the Future?*

## SEMESTER VIII: FUNDAMENTALS OF PYTHON PROGRAMMING

Course Type:	<b>Minor</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 20, Practical: 30, Summative: 70)</b>

### Course Objectives:

- To get introduced to Python programming.
- To provide the concept of flow control, functions and its implementation in Python.
- To provide the ability to design object – oriented programs with Python.
- To provide the knowledge of file processing in Python.

### Course Outcome:

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

- learn Python installation, Anaconda installation and work with Google Colab, Jupyter Notebook, and Spyder.
- define and demonstrate the use of built-in data structures like lists, tuples and dictionary
- use functions and modules in Python programming.
- design object- oriented programs and to handle files in Python.

### Unit 1: Installation of Python

Python Installation, Working with Google Colab, Anaconda Installation, Working with Jupyter Notebook, Spyder

### Unit 2: Introduction to Python

Python, Python Interpreter, Using Python as calculator, Python shell, Indentation.

### Unit 3: Python Basics

Atoms, Identifiers and keywords, Literals, Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment operator, Bit wise operator, Identity and Membership operator.

### UNIT 4: Strings

Strings, String properties, Built-in functions, String Methods, String Conversions, String Comparisons.

### UNIT 5: Decision Control Statements

Input and Output Statements, Control statements, Conditional Statements- if...else, Nested if statements, Ternary Operators.

### UNIT 6: Loop Control Statements

Looping- while Loop, for Loop, Loop Control, break, continue and pass.

### UNIT 7: Lists

Lists, Accessing List elements, looping in Lists, Basic List operations, Using built-in functions on Lists, List methods, Sorting and reversing.

### UNIT 8: Tuples

Tuples, accessing tuple elements, looping in tuples, basic tuple operations, using built-in functions on

Tuples, Tuple methods, Tuple varieties

#### **UNIT 9: Sets**

Definition of set, accessing set elements, looping in sets, basic set operations, Sets, Sets Methods and Functions.

#### **UNIT 10: Dictionaries**

Dictionary, definition of dictionaries, accessing dictionary elements, looping in dictionaries, basic dictionary operations.

#### **UNIT 11: Functions**

Functions: Definition, call, positional and keyword parameter. Default parameters, variable number of arguments.

#### **UNIT 12: Modules**

Modules: import mechanisms, Functional programming: map, filter, reduce, max, min. lambda function.

#### **UNIT 13: Object Oriented Programming**

Object Oriented Programming, Definition of Classes and Objects, Python Inheritance, Different types of Inheritance with examples.

#### **UNIT 14: File Processing**

File Processing: reading and writing files, manipulating file pointer using seek

#### **References:**

1. Budd, T. (2010). *Exploring Python*. McGraw Hill Higher Education.
2. [The Python Tutorial — Python 3.11.2 documentation](#)
3. Wentworth, P., Elkner, J., Downey, A. B., & Meyer, C. (2015). *How to think like a computer scientist: Learning with Python 3*.

## **SEMESTER VII: RESEARCH METHODOLOGY**

Course Type:	<b>SEC</b>
Number of Credits:	<b>4</b>
Total Marks:	<b>100 (Formative: 30, Summative: 70)</b>

#### **Course Objectives:**

- To provide students with an understanding of the research process and the steps involved in conducting research, as well as the importance of research in computer science.
- To equip students with the skills and knowledge necessary to formulate research problems, develop research questions and hypotheses, and conduct a thorough literature review.
- To familiarize students with various research designs, sampling techniques, and data collection and analysis methods, and enable them to choose appropriate methods for their research projects. Additionally, students will learn how to interpret and present their research findings in a clear and effective manner.

#### **Course Outcome:**

After going through this unit, learner will be able to

- understand the importance of research in computer science and the various types of research that can be conducted.
- develop skills to identify research problems, formulate research questions and hypotheses, and conduct a literature review to support research.
- acquire knowledge on research design, sampling techniques, data collection methods, and data analysis techniques, and apply them to conduct and interpret research.

**Unit 1: Introduction to Research Methodology**

Definition and types of research, Importance of research in computer science, Research process and steps involved

**Unit 2: Research Problem Formulation**

Identifying research problems, Formulating research questions and hypotheses, Literature review and its significance

**Unit 3: Research Design**

Types of research designs, Choosing appropriate research design, Validity and reliability in research

**Unit 4: Sampling Techniques**

Types of sampling techniques, Sampling methods and their suitability, Sampling error and sample size determination

**Unit 5: Data Collection Methods**

Primary and secondary data collection methods, Survey, questionnaire, and interview methods, Data collection tools and techniques

**Unit 6: Data Analysis Techniques**

Quantitative data analysis techniques, Qualitative data analysis techniques, Statistical analysis and software packages

**Unit 7: Interpretation of Results**

Interpreting quantitative and qualitative data, Drawing conclusions and making recommendations, Presenting results in written and graphical form

**Unit 8: Writing a Research Proposal**

Components of a research proposal, Writing a proposal for a research project, Guidelines for research proposal writing

**Unit 9: Ethics in Research**

Ethical considerations in research, Institutional review board (IRB) and its role, Ethical guidelines for research

**Unit 10: Research Report Writing**

Structure and format of research reports, Preparing a research report, Writing skills for research reports

**Unit 11: Presentation Skills**

Oral presentation skills, Visual aids for presentation, Effective communication skills

**Unit 12: Project Management**

Planning and managing research projects, Time management techniques, Resource allocation and budgeting

**Unit 13: Case Study Research**

Definition and characteristics of case study research, Types of case studies, Analyzing case study data

**Unit 14: Experimental Research**

Definition and characteristics of experimental research, Types of experimental research, Conducting and analyzing experimental research

**References:**

1. Research Methodology: A Step-by-Step Guide for Beginners by Ranjit Kumar
2. Research Methods in Computer Science by Erika Ábrahám, Catuscia Palamidessi, and Ana Sokolova

## SEMESTER VII: SEMINAR

Course Type:	SEMINAR
Number of Credits:	4
Total Marks:	100

### **Guidelines for presenting seminar paper**

- 1. Choose a topic:** The first step is to choose a topic for your seminar paper presentation. The topic should be relevant to your BCA program and should be something that you are interested in researching.
- 2. Research your topic:** Once you have chosen your topic, you need to research it thoroughly. Use academic sources such as journals, books, and reputable websites to gather information.
- 3. Create an outline:** Create an outline of your seminar paper presentation. This should include an introduction, a literature review, methodology, findings, discussion, and conclusion.
- 4. Prepare your slides:** Once you have created an outline, prepare your slides. Your slides should be easy to read and should contain bullet points and visuals to help your audience understand your presentation.
- 5. Start with an introduction:** Begin your presentation with a brief introduction to your seminar paper topic, including its relevance and significance to the field of computer applications.
- 6. Use visuals:** Use visuals like charts, diagrams, and graphs to help illustrate the concepts you are presenting. This will make your presentation more engaging and easier to understand.
- 7. Be concise:** Make sure to be concise and clear in your presentation, avoiding unnecessary jargon or technical terms that your audience may not be familiar with.
- 8. Explain technical terms:** If you must use technical terms, make sure to explain them in simple language that your audience can understand.
- 9. Practice your delivery:** Practice your presentation beforehand so that you are comfortable with your delivery and can maintain eye contact with your audience.
- 10. Engage with your audience:** Encourage questions and engagement from your audience throughout your presentation, and be prepared to answer any questions they may have.
- 11. Conclude with a summary:** Conclude your presentation with a brief summary of your key points and reiterate the main takeaways from your seminar paper.

By following these guidelines, you can deliver an effective and engaging seminar paper presentation for your BCA programme.

# ANNEXURE III

## PROJECT GUIDELINES FOR BCA PROGRAMME

### SEMESTER VIII: PROJECT

Course Type:	<b>PROJECT</b>
Number of Credits:	<b>12</b>
Total Marks:	<b>200</b>

#### A. PROJECT GUIDE ELIGIBILITY CRITERIA:

Full Time Faculties in the Department of Computer Science/ Information Technology of KKHSOU/ Colleges/ Institutions affiliated to any Indian University recognized by UGC and having minimum 2 years teaching experience.

OR

A person having minimum M.Tech., MCA, M.Sc. in Computer Science/Information Technology from a UGC recognized university with 4 years of experience in IT sector.

#### B. TYPE OF PROJECT

Learner may choose any topic according to BCA standard. Most of the project work falls under the following types:

- a. Database oriented (e.g. payroll system, Loan management system etc.)
- b. Application oriented (e.g. Mobile apps development, web based development)
- c. R & D project (e.g. Image processing, speech processing, data mining, networking, expert systems etc.)

#### C. PROJECT PROPOSAL (SYNOPSIS)

A project proposal or synopsis is a crucial framework that outlines the necessary details about a project. It provides a roadmap that helps project stakeholders understand the objectives, methodology, and expected outcomes of the project.

The necessary parts of a project proposal, as mentioned in the given format, help to structure the proposal and provide a comprehensive overview of the project.

- i. Title of the Project: [Insert the title of your project here]
- ii. Introduction and Objectives of the Project: [Provide a brief introduction to



- your project and describe the objectives you aim to achieve through it.]
- iii. Project Category: [Specify the category of your project - RDBMS, application development, research and development, etc.]
  - iv. Tools, Platform, Hardware and Software Requirement Specifications: [List the tools, platforms, hardware, and software required for carrying out your project.]
  - v. Whether the Project is Done for Any Industry/Client? The Name and Address of the Industry or Client is to be Mentioned: [Specify whether your project is being developed for a particular industry or client. If yes, mention the name and address of the industry or client.]
  - vi. Methodology and Expected Output: [Provide details of the methodology you will be using to carry out your project and describe the expected output.]
  - vii. Conclusion: [Summarize your project proposal and highlight its significance.]

Consultation with a project guide is crucial to ensure that the project proposal/synopsis meets the required standards and expectations. A project guide or supervisor has significant experience and knowledge in the respective field, and their feedback can help refine the proposal and identify any gaps or potential issues that need to be addressed before the project begins.

Moreover, project guides can provide valuable insights and feedback that can enhance the quality of the project and improve the chances of its success. Therefore, it's essential to involve your project guide from the outset and maintain regular communication throughout the project's lifecycle to ensure that it's on track and aligned with the overall objectives.

#### **D. APPLICATION AREAS & RELATED TOOLS**

The following is a list of application areas and related tools that can be considered for developing a project. The chosen area will depend on the project's objectives, scope, and requirements.

##### **Applications Areas:**

- Database Management System
- Computer Graphics
- Internet/ Intranet
- Computer Networking-Communication Software
- E-Commerce
- TCP/IP/Routing protocols

- Image processing
- Speech Processing
- Web Application Development
- Mobile application development etc.

**Related Tools:**

- **FRONT END / GUI Tools:** PHP, JavaScript, HTML, Scripting languages etc.
- **BACKEND:** Python, PHP, JavaScript etc.
- **RDBMS:** Oracle, MYSQL, No SQL, MongoDB etc.
- **LANGUAGES:** Python, Matlab, Scilab, C, C++, Java, C# etc.
- **INTERNET TECHNOLOGIES:** Java, Active X, SWING, JSP, ASP, PHP, XML etc.
- **OPERATING SYSTEMS:** WINDOWS/ LINUX /MAC OS/ANDROID.
- This comprehensive list covers a wide range of areas and tools that offer diverse opportunities for project development. However, the selection of a specific area and related tools will depend on the project's objectives and requirements.

**E. PROJECT REPORT GUIDELINE**

The Project report should be prepared in well-structured preferably typed in Latex. Depending on the type of project the report should be as follows:

**(a) For Database project:**

- i. Declaration: This page should include a statement declaring that the work presented in the report is the student's original work.
- ii. Certificate: This page should include a certificate from the institution confirming that the student has completed the project.
- iii. Certificate from Guide: This page should include a certificate from the project guide confirming that the student has completed the project under their supervision.

**CHAPTER I: INTRODUCTION**

- 1.1 Brief idea about the project
- 1.2 Objective of the project
- 1.3 Scope of the project
- 1.4 Existing system
- 1.5 Proposed System
- 1.6 Platform used(Hardware & Software)
- 1.7 Project location

## **CHAPTER II: REQUIREMENT ANALYSIS**

- 2.1 Introduction
- 2.2 Tools used for Requirement gathering
- 2.3 Problem in Existing System
- 2.4 Conclusion

## **CHAPTER III: LOGICAL DESIGN**

- 3.1 Introduction
- 3.2 DFD (0<sup>th</sup>, 1<sup>st</sup>, 2<sup>nd</sup> level)
- 3.3 ER diagram
- 3.4 Use case diagram
- 3.5 Activity diagram
- 3.6 Conclusion

## **CHAPTER IV: PHYSICAL DESIGN**

- 4.1 Introduction
- 4.2 Database Design (Give your normalized database here)
- 4.3 Module design
- 4.4 Input/output design
- 4.5 Conclusion

## **CHAPTER V: IMPLEMENTATION**

- 5.1 Introduction
- 5.2 Process description (if any)
- 5.3 Output & Report
- 5.4 Conclusion

## **CHAPTER VI: TESTING**

- 6.1 Introduction
- 6.2 Types of testing performed
- 6.3 Conclusion

References Appendix (if any)

**(b) R & D/ Application project:**

- Acknowledgement
- Content with page number
- Declaration Certificate
- Certificate from Guide

## **CHAPTER I: INTRODUCTION**

- 1.1 Brief idea about the project
- 1.2 Objective of the project
- 1.3 Scope of the project
- 1.4 Application of the project
- 1.5 Proposed System
- 1.6 Platform used(Hardware & Software)
- 1.7 Project location

## **CHAPTER II: LITERATURE REVIEW**

- 2.1 Introduction
- 2.2 Work already done in the area(Historical evidence)
- 2.3 Problem in Existing technology
- 2.4 Conclusion

## **CHAPTER III: THEORITICAL BACKGROUND**

- 3.1 Introduction
- 3.2 Theory used in the project
- 3.3 Conclusion

## **CHAPTER IV: RESULT AND DISCUSSION**

- 4.1 Introduction
- 4.2 Methodology
- 4.3 Result
- 4.4 Analysis on result
- 4.5 Conclusion

## **CHAPTER V: Conclusion & Future work**

- 5.1 Introduction
- 5.2 Chapter wise conclusion
- 5.3 Future work

References Appendix (if any)

## CERTIFICATE OF ORIGINALITY FROM THE PROJECT SUPERVISOR

This is to certify that the project report entitled .....

.....

submitted to **Krishna Kanta Handiqui State Open University** in partial fulfilment of the requirement for the award of the degree of **BACHELOR OF COMPUTER APPLICATIONS (BCA)**, is an original work carried out by Mr./Ms.....

**Enrolment No.:** ..... under the supervision of Dr./Mr./Ms.....

The matter embodied in this project is a genuine work done by the student and has not been submitted either to this University or to any other University/Institute for the fulfilment of the requirement of any course of study.

Signature of the Learner

Signature of the Project Supervisor

Name:  
Address:  
Enrolment No.:

Name:  
Designation:  
Address

