

**Three/Four Yr. Degree Course
(Single Major & Double Minor with Research)**

based on NEP-2020

COMPUTER SCIENCE

EFFECTIVE FROM 2026-27



SAMBALPUR UNIVERSITY

JYOTI-VIHAR, BURLA, SAMBALPUR, ODISHA-768019

The model curriculum and credit frame work for under graduate programmes as multiple entry and exit option with research and without research of National Education Policy-2020.

CERTIFICATE COURSE: One Academic Year (First & Second Semester and Summer Vocational Course and Community Work) 44 CREDITS

DIPLOMA COURSE: Two Academic year (First, Second, Third & Fourth Semester and Summer Vocational Course and Community Work) 86 CREDITS

3YR. DEGREE COURSE: Three Academic Year (First, Second, Third, Fourth, Fifth & Sixth Semester and Community Work) 126 CREDITS

4YR. DEGREE COURSE (HONOURS): Four Academic Year (First, Second, Third, Fourth, Fifth, Sixth, Seven & Eight Semester and Community Work) 166 CREDITS

4YR. DEGREE COURSE (HONOURS WITH RESEARCH): Four Academic Year (First, Second, Third, Fourth, Fifth, Sixth, Seven & Eight Semester and Community Work) 166 CREDITS

Type of Course:

Major (Core-I): is the subject of main focus and the degree will be awarded in that discipline.

Minor Stream (Core-II): helps a student to gain a broader understanding knowledge beyond the major discipline. A student may gather 28 credits from a single subject

Multidisciplinary U.G. Programme (09 Credits): Three introductory-level Multidisciplinary Course baskets are offered for students to choose any one from each semester outside of the subject/discipline/faculty: A group of related subject i.e., faculty (Arts/Commerce/Science). The student can opt three different subject/course for each semester examination.

Ability Enhancement Course (AEC 08 Credits): Student are required to achieve competency in a Modern Indian Language (MIL) and in the English language with special emphasis on language and communication skills (not literally skill). Literature and Linguistic courses shall be offered in major core and minor stream. In lieu of Odia, a student may opt for only Sanskrit, Hindi or Urdu.

Skill Enhancement Course (SEC 09 Credits): A student has to opt the Skill Enhancement courses from the basket as per the subject available in the institution. The students can opt for any one course for each semester examination and these courses are aimed at imparting practical skills, hands-on training, soft skills, etc., to enhance the employability of students.

Value Added Course (VAC 12 Credits): The VAC course basket are prearranged for the students to choose for each semester's examination.

Summer Internship/Apprenticeship/Summer Project (04 Credits with 120 Hours): Community Engagement and Services/ Field work/ Internship programme is mandatory for all students of 4th Semester undergraduate Programmes of NEP-

2020 carrying 4 credits which shall be completed in 120h (Students admitted in the session 2024-25 onwards) under affiliated colleges of Sambalpur University.

Internship: A course requiring students to participate in a professional activity or work experience or cooperative education activity with an entity external to the education institution, normally under the supervision of an expert of the given external entity. A key aspect of the internship is induction into actual work situations.

Internships may be undertaken anywhere in India, subject to institutional approval and compliance with credit requirements.

Internships involve working with local industry, government or private organizations, business organizations, artists, crafts persons, NGOS, Private Sector organizations, Government / Public Sector institutions, Industries, MSMEs (Online or Offline) and similar entities to provide opportunities for students to actively engage in on-site experiential learning.

Online / Digital Internship Opportunities: Internship may also be undertaken in online or hybrid mode through recognized Government of India platforms, subject to institutional approval and fulfilment of credit requirements.

Students may undertake internships or field work through their own arrangement, subject to prior approval of the concerned HEI.

Community Engagement and Services/ Field work: Community Engagement activities shall be conducted through NSS. Students not enrolled under NSS may undertake community engagement projects / field work for 120h duration under the supervision of faculty members.

After completion of Internship students will collect an internship completion certificate from the organization /competent authority. Submission of Report and Evaluation: Students opted for Internship/ Community engagement projects/ Field work are required to submit a report to the Institution for evaluation. Supporting documents like feedback from the host Organisation, geotagged photos and other relevant documents should be submitted with the report.

Table-I: Course Table Structure for Certificate Course: Exit after first Academic Year One Academic Year: (1st & 2nd Semester and a Summer Vocational Course and Community

OUTLINE OF SYLLABUS
B.Sc. COMPUTER SCIENCE
Course Structure

Semester	Subjects	Credit
I	Major-I-P-I - Introduction to Computer Science and Python Programming	4
	Major-I-P-II- Problem Solving with C Programming	4
	Minor-I-P-I- Problem Solving with C Programming	4
	MDC-1-Computer Fundamentals	3
	AEC-1-Odia/Hindi/Sanskrit (Compulsory)	4
	VAC-1-Environmental Studies and Disaster Management (Compulsory)	3
II	Major-1-P -III- Data Structures	4
	Major-1-P - IV - Data Base Management System	4
	Minor-II-P-I- Data Structures	4
	MDC-II- Introduction to Web Technologies	3
	AEC-II-English (Compulsory)	4
	SEC-1-Advanced Python Programming	3
III	Major-1-P -V-Discrete Mathematical Structures	4
	Major-1-P - VI- Computer Organization	4
	Major-1-P -VII-Operating Systems	4
	Minor-I-P-II - Operating Systems	4
	MDC-III-Internet and Ethical Practices	3
	VAC II- VAC basket of other Schools	3
IV	Major-1-P -VIII- Web Technologies	4
	Major-1-P- IX- Object Oriented Programming using C++	4
	Major-1-P- X- Computer Network	4
	Minor-II-P-II- Computer Network	4
	Internship	4
V	Major-1-P -XI- Software Engineering	4
	Major-1-P - XII- (A) Introduction to Artificial Intelligence (B) Computer Graphics	4
	Major-1-P -XIII- Programming in Java	4
	Minor-I-P-III -Introduction to Artificial Intelligence	4
	SEC-II-Principles of Management	3
	VAC III- VAC basket of other Schools	3
VI	Major-1-P - XIV- Algorithm Design Techniques	4
	Major-1-P - XV- Project Work-I	4

	Minor-II-P-III- Algorithm Design Techniques	4
	SEC-III-SEC basket of other Schools	3
	VAC IV- VAC basket of other Schools	3

B.Sc. COMPUTER SCIENCE (Without Research)		
Semester	Course Title	Credit Hour
VII	Major-1-P - XVI- Theory of Computation	4
	Major-1-P - XVII- Computer Architecture	4
	Major-1-P -XVIII- Mathematics for Machine Learning	4
	Major-1-P - XIX-(A) Data Mining or (B) Wireless Sensor Networks or (C) Cyber Security	4
	Minor-1-P-IV- Cyber Security	4
VIII	Major-1-P -XX- Cloud Computing	4
	Major-1-P - XXI- Machine Learning	4
	Major-1-P -XXII- (A) Compiler Design or (B) Internet of Things or (C) Introduction to Data Science	4
	Major-1-P-XXIII- Project Work-II	4
	Minor-1-P-V- Introduction to Data Science	4

B.Sc. COMPUTER SCIENCE (With Research)		
Semester	Course Title	Credit Hour
VII	Major-1-P - XVI- Theory of Computation	4
	Major-1-P - XVII- Computer Architecture	4
	Major-1-P -XVIII- Mathematics for Machine Learning	4
	Minor-1-P-IV-Computer Architecture	
	Minor-1-P-V-Mathematics for Machine Learning	
VIII	Major-1-P -XIX- Cloud Computing	4
	Major-1-P - XX- Machine Learning	4
	Major-1-P -XXI- (A) Compiler Design or (B) Internet of Things or (C) Introduction to Data Science	4
	Major-1-P-XXII- Research Methodology	4
	Major-1-P-XXIII - Dissertation	8

MULTIDISCIPLINARY COURSES

Semester	Subjects	Credit
I	A. Computer Fundamental	3
II	B. Introduction to Web Technologies	3
III	C. Internet and Ethical Practices	3

SKILL ENHANCEMENT COURSES (SEC)

Semester	Subjects	Credit
II	A. Advanced Python Programming	3
V	B. Principles of Management	3
VI	C. Students will choose SEC basket of other Dept./Schools	3

VALUE AIDED COURSES (VAC)

Semester	Subjects	Credit
I	A. Environmental Studies and Disaster Management	3
III	B. Digital Fluency	3
V	C. Intellectual Property Right (IPR)	3
VI	D. Social Statistics and Computer Applications	3

VOCATIONAL COURSES

Semester	Subjects	Credit
I	A. Office Automation Tools	3
III	B. Multimedia and Animation	3
V	C. Introduction to Web Design	3
VI	D. Data Analysis using Spread sheet	3

**B.Sc. COMPUTER SCIENCE
(Without Research)**

Semester	Core-I	Core-II	Core-III	Multi Disciplinary Course (MDC)	AEC	SEC	VAC	Community Engagement & Services/ Field work /Internship	Total Minimum Credit
I	2X4=8 1. Introduction to Computer Science and Python Programming 2. BCS Problem Solving with C Programming	1X4=4 Problem Solving with C Programming		1X3=3 Computer Fundamentals	1X4=4 Odia/Hindi/Sanskrit (Compulsory)		1x3=3 Environmental Studies and Disaster management		22
II	2X4=8 1. Data Structures 2. Data Base Management System		1X4=4 Data Structures	1X3=3 Introduction to Web Technologies	1X4=4 English	1X3=3 Advanced Python Programming			22
Vocational Course of 4 credits for Certificate									44
III	3X4=12 1. Discrete Mathematical Structures 2. Computer Organization 3. Operating Systems	1X4=4 Operating Systems		1X3=3 Internet and Ethical Practices			1X3=3 VAC basket of other Schools		22
IV	3X4=12 1. Web Technologies Object 2. Oriented Programming using C++ 3. Computer Network		1X4=4 Computer Network					1x4=4 Internship	20
Vocational Course of 4 credits for Diploma Certificate									42
V	3X4=12 1. Software Engineering 2. A-Introduction to Artificial Intelligence	1X4=4 Introduction to Artificial Intelligence				1x3=3 Principles of Management	1x3=3 VAC basket of other Schools		22

	B - Computer Graphics 3.Programming in Java								
VI	2X4=8 1. Algorithm Design Techniques 2. Project Work-I (Major-15)		1X4=4 Algorithm Design Techniques			1x3=3 SEC basket of other Schools	1x3=3 VAC basket of other Schools		18
									40
VII	4x4=16 1. Theory of Computation 2.Computer Architecture 3.Mathematics for Machine Learning 4. A-Data Mining B: Wireless Sensor Networks C: Cyber Security	1x4=4 Cyber Security							20
VIII	4x4=16 1.Cloud Computing 2.Machine Learning 3.A: Compiler Design B: Internet of Things C: Introduction to Data Science 4.Project Work-II(For 4-Year Hons. without Research)	1x4=4 Introduction to Data Science							20
									40
Total	23x4=92	5x4=20	3x4=12	3x3=9	2x4=8	3x3=9	4x3=12	1x4=4	166

**B.Sc. COMPUTER SCIENCE
(With Research)**

Semester	Core-I	Core-II	Core-III	Multi-disciplinary	AEC	SEC	VAC	Community Engagement & Services/Field work/Internship	Total Minimum Credit
I	2X4=8 1. Introduction to Computer Science and Python Programming 2. BCS Problem Solving with C Programming	1X4=4 Problem Solving with C Programming		1X3=3 Computer Fundamentals	1X4=4 Odia/Hindi/Sanskrit (Compulsory)		1x3=3 Environmental Studies and Disaster management		22
II	2X4=8 1. Data Structures 2. Data Base Management System		1X4=4 Data Structures	1X3=3 Introduction to Web Technologies	1X4=4 English	1X3=3 Advanced Python Programming			22
Vocational Course of 4 credits for Certificate									44
III	3X4=12 1. Discrete Mathematical Structures 2. Computer Organization 3. Operating Systems	1X4=4 Operating Systems		1X3=3 Internet and Ethical Practices			1X3=3 VAC basket of other Schools		22
IV	3X4=12 1. Web Technologies 2. Object Oriented Programming using C++ 3. Computer Network		1X4=4 Computer Network					1x4=4 Internship	20
									42
V	3X4=12 1. Software Engineering 2. A-Introduction to Artificial	1X4=4				1x3=3	1x3=3		22

	Intelligence B-Computer Graphics 3.Programming in Java								
VI	2X4=8 1. Algorithm Design Techniques 2. Project Work-I (Major-15)		1X4=4 Algorithm Design Techniques			1x3=3 SEC basket of other Schools	1x3=3 VAC basket of other Schools		18
									40
VII	3x4=12 1. Theory of Computation 2. Computer Architecture 3. Mathematics for Machine Learning	2x4=8 1. Computer Architecture 2. Mathematics for Machine Learning							20
VIII	2x4=8 1. Cloud Computing 2. Machine Learning							Research 12 (4+8) (Research Methodology: 4 credits + Dissertation: 8 credits)	20
									40
Total	20x4=80	5x4=20	3x4=12	3x3=9	2x4=8	3x3=9	4x3=12	16	166

SEMESTER-I

Major-I-P-I : Introduction to Computer Science and Python Programming

Course Outcomes:

- To learn the fundamentals of computer and its working mechanism.
- To learn programming paradigms and design.
- To learn the basics of Python Programming for problem solving.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. gain knowledge about the components of a computer with their functions
2. have an exposure to various computer networks
3. learn the steps for program development
4. learn the use of control structures, built-in functions, lists, recursions in Python programming

Unit-I:

Introduction to Computers: Generation of computers, types of computers. Computer System hardware: Block diagram, CPU, Memory unit, and Storage devices with their functions. Computer memory: Memory hierarchy, CPU Registers, Cache Memory, Primary Memory, Secondary memory, access methods.

Input & Output Units: Various types of Input and Output devices with their functions. Computer Software:

Introduction, Types of software: System software, Application software, Utility software, firmware and their usage.

Computer Network: Importance of Networking, Network types: LAN, MAN, WAN, Internet and its applications, network devices: Repeater, bridge, hub, switch, router, and Gateway.

Unit-II:

Program Development: Problem Analysis, program design, development. Algorithm: Introduction, Pseudo code, Control Structures. Flowchart: Introduction, Symbols, preparing a flowchart. Programming Paradigms & Design: Structured programming, Object-oriented programming, Top-down and Bottom-up design approaches. Characteristics of a good program.

Programming Languages: Concept of Low-level & high-level languages, Different generations of Programming Languages. Translators: Assembler, Compiler, Interpreter. Linker, Loader, Editors, Integrated Development Environment. Programming Errors: Syntax, Semantic, Logical, Compile-time, run-time, link-time, environmental, Input/output errors. testing and debugging.

Unit-III:

Introduction to Python, getting started with Python, Python Basics: Identifiers, Keywords, Python types, basic types, mutable and immutable types, Integer & float ranges, Variable type & assignment, Arithmetic Operators, Precedence & Associativity, Conversions, built-in functions, modules, container types, comments & indentation, multi-lining, classes & Objects, Multiple Objects. Strings: Introduction, Accessing String elements, Properties, built-in functions, Methods, Conversions, Comparisons.

Console I/O: I/O operations, formatted printing. Decision Control Instruction: Logical operators, Conditional Expressions, all () & any (), receiving input, pass statement. Repetition Control Instruction: types, usage of loops, break & continue, else block of a loop.

Unit-IV:

Lists, Sets, Tuples, Dictionaries: creating, accessing, and looping in each type. Applying basic operations, using built-in functions and methods on each type, possible data structure / mathematical operations on each type.

Comprehensions on List, Set, and dictionary. Functions: built-in and user-defined functions, invoking functions, unpacking arguments. recursive function, iteration vs recursion.

Text Books:

- ✓ *Computer Fundamentals* by Anita Goel, Pearson Pub.
- ✓ *Let us Python* by Yashavant Kanetkar & Aditya Kanetkar, BPB Pub.

Reference Books & e-Resources:

- ✓ *Fundamentals of Computers* by Reema Thareja, Oxford University Press
- ✓ *Computer Fundamentals* by D. P. Nagpal, S. Chand Pub.

Core-I Lab: Programming using Python

1. Write a program to demonstrate the usage of various arithmetic operators.
2. Write a program that makes use of trigonometric functions available in math module.
3. Write a program that will convert various temperatures.
 - a. Fahrenheit to Centigrade
 - b. Centigrade to Fahrenheit
4. Write a program that will find the roots of a quadratic equation: $ax^2 + bx + c = 0$

5. Write a program that demonstrate the usage of various String functions.
6. Write a program that will ask you to enter your name, through keyboard, and perform following operations
 - a. Find the middle name
 - b. Find the last name (using string slicing)
 - c. Re-write the name with surname first.
7. Write a program to find out whether the integer entered by the user, through the keyboard, is even or odd number.
8. Find out the youngest among Shyam, Dugu and Ishan whose ages are entered by the user through keyboard.
9. Given three points (x1, y1), (x2, y2), (x3, y3), write a program to check all the three points fall on one straight line.
10. Write a program that will print the odd numbers from n1 to n2 where the values of n1 and n2 are entered by the user.
11. Write a program to find the factorial value of a number entered by the user.
12. Write a program to print all prime numbers between n1 to n2 where the values of n1 and n2 are entered by the user.
13. Write a program to demonstrate basic operations on the list.
14. Write a program to demonstrate stack and queue operations using a list of numbers.
15. Write a program to ask the data of five students that contain name, roll number, age. Sort the list based on roll number of the Student. [Note: Use list of lists].
16. Write a program that will add two square matrices. The dimension and elements of the matrices will be entered by the user.
17. Write a program to demonstrate basic operations on the tuple.
18. Store the data about the shares held by the user as tuples containing the following information about shares: share name, cost price, number of shares, selling price. Write a program to determine:
 - a. total cost of the portfolio
 - b. total amount gained or lost
19. Write a program to demonstrate basic operations on the set.
20. Create an empty set. Write a program that adds five student names to this set, modifies one existing name, and deletes two names existing in it. [ask the user which name to modify/delete].
21. Write a program to demonstrate basic operations on the dictionary.
22. Create a dictionary to store data (name, roll number) of N students. The key will be the roll number of the student and the value contains the data of the student (in a list). Write a program that asks the user to enter a name of a Student, search it in the dictionary and print the data of the Student if it is available otherwise display an appropriate message.
23. Write a program to demonstrate basic comprehensions on list, set and dictionary.
24. Write a program that will find x^n (x to the power of n) using a function. The function receives the value of x , n and should return the value of x^n . [don't use any mathematical function].

Major-I-P-II / Minor-I-P-I : Problem Solving with C Programming

Course Objectives:

- To learn the C programming language to solve different scientific and business problems
- To learn how to design and write effectively codes using various programming constructs available in the C programming language

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Gain knowledge about different data types and operators in C language
2. Learn the use of various control structures and array
3. Learn the use of pointers, functions, and storage classes
4. Write programs using structures, union, and files

Unit-I:

Introduction: Introduction to Programming Language, Introduction to C Programming, Keywords & Identifiers, Constants, Variables, Input and Output Operations, Compilation and pre-processing, Data types: Different data types, Data types qualifier, modifiers, Memory representation, size and range, Operators: Operators (Arithmetic, Relational,

Logical, Bitwise, Assignment & compound assignment, Increment & Decrement, Conditional), Operator types (unary, binary, ternary). Expressions, Order of expression (Precedence and associativity)

Unit-II:

Decision Control structures & Loops: Decision Making and Branching statements (Simple IF, IF...ELSE, Nested IF...ELSE, ELSE ... IF ladder), Selection control structure (Switch Statement). Looping statements (FOR, WHILE, DO...WHILE), break, continue and GOTO statements

Array: Concept of Array, Array Declaration, types of array (one and multiple dimension), Character Arrays and Strings, limitation of array.

Unit-III:

Pointers: Concept of Pointer (NULL pointer, wild pointer, dangling pointer, generic pointer), Pointer Expressions, Accessing the Address of a Variable, Declaring Pointer Variables, Initializations of Pointer Variable, Accessing a Variable through its Pointer, Pointer arithmetic, Pointer representation of array, Array of Pointers, Accessing String using Pointer.

Function: Types of Function, Function Declaration, Function Definition, Function Call, Recursive Function, Dynamic Memory Management functions, String handling function (strlen, strcmp, strcpy, strncpy, strcat, strstr).

Storage class: Types (auto, register, static, extern), scope rules, declaration and definition.

Unit-IV:

Structure and Union: Defining, Declaring, Accessing, Initialization Structure, nested structure, self-referential structure, bit-field, Arrays of Structures, Structures and Functions, structures and pointers, Unions, difference between structure and union, structure within union. File: File Management in C, Defining and Opening a File, File opening modes (read, write, append), Closing a File, File operations, Error handling during I/O Operations, sequential and random access files. Command line arguments.

Text Book:

Programming in ANSI C by E. Balagurusamy, TMH

Reference Books:

- ✓ *The C Programming Language by B. Kernighan & Dennis Ritchie, PHI.*
- ✓ *C: How to Program by Paul Deitel, Harvey Deitel, Prentice Hall.*
- ✓ *Programming using C by P.C. Sethi & P.K. Behera, Kalyani Publisher.*

Major-I-P-II / Minor-I-P-I - Lab: Problem Solving with C Programming

1. Write a Program to find greatest among three numbers.
2. Write a Program to all arithmetic operation using switch case.
3. Write a Program to print the sum and product of digits of an integer.
4. Write a Program to reverse a number.
5. Write a Program to compute the sum of the first n terms of the following series
 $S = 1 + 1/2 + 1/3 + 1/4 + \dots$
6. Write a Program to compute the sum of the first n terms of the following series
 $S = 1 - 2 + 3 - 4 + 5 - \dots$
7. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
8. Write a function to find whether a given number is prime or not. Use the same to generate the prime numbers less than 100.
9. Write a Program to compute the factors of a given number.
10. Write a program to swap two numbers.
11. Write a Program to print a triangle of stars as follows (take number of lines from user):
*

12. Write a Program to perform following actions on an array entered by the user:
 - a) Print the even-valued elements
 - b) Print the odd-valued elements
 - c) Calculate and print the sum and average of the elements of array
 - d) Print the maximum and minimum element of array
 - e) Remove the duplicates from the array
 - f) Print the array in reverse order

The program should present a menu to the user and ask for one of the options. The menu should also include

options to re-enter array and to quit the program.

13. Write a Program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
14. Write a program that swaps two numbers using pointers.
15. Write a program in which a function is passed address of two variables and then alters its contents.
16. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.
17. Write a program to find sum and average of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc() functions.
18. Write a menu driven program to perform following operations on strings:
 - a) Show address of each character in string
 - b) Concatenate two strings without using strcat function.
 - c) Concatenate two strings using strcat function.
 - d) Compare two strings
 - e) Calculate length of the string (use pointers)
 - f) Convert all lowercase characters to uppercase
 - g) Convert all uppercase characters to lowercase
 - h) Calculate number of vowels
 - i) Reverse the string
19. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
20. Write a program to copy the content of one file to other.

SEMESTER-II

Major-1-P -III / Minor-II-P-I: Data Structures

Course Objectives:

- To understand different ways of organizing data in computer's memory.
- To learn different operations on data structures.
- To explore different applications of data structures.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn about data structures and the use of array
2. Create linked lists and perform insertion/deletion operations on them
3. Represent Stack and Queue in the memory and learn their applications
4. Learn the use of various non-linear data structures and their applications

Unit-I:

Introduction to Data Structures: Definition, Concepts, Classification of Data Structures. Array: Introduction, One-Dimensional Array, Memory representation, Operations: Traversing, Searching, Insertion, Deletion, Merge. Two-Dimensional Array & Memory Representation, Multidimensional Array. Linear Search versus Binary Search, Sorting: Selection Sort, Bubble Sort.

Unit-II:

Linked Lists: Definition, Single Linked List, Memory representation, Operations: Traversing, Searching, Insertion, Deletion and Merge. Double Linked List, Operations: Insertions, Deletion. Circular, Double Circular Linked list, Operations: Traversing, Insertion. Applications of Linked List, Sparse Matrix and Polynomial representations.

Unit-III:

Stack: Definition, Representation: Array and Linked List representations, Operations: PUSH, POP, STATUS. Applications: Evaluation of Arithmetic Expressions: Notations, Infix to Postfix Conversion, Evaluation of Postfix expression. Recursion (Factorial and Fibonacci), Tower of Hanoi.

Queues: Definition, Representation: Array and Linked List representations, Operations: Enqueue, Dequeue. Structures of Queue: Circular, Deque and Priority Queue. Applications of Queue

Unit-IV:

Trees: Definition, Terminologies, Binary Tree: Properties, Representations (Linear and Linked List representations). Operations: Traversal (Inorder, Preorder, Postorder), Search. Introduction to Binary Search Tree, AVL tree, M-Way Search Tree. Applications of Trees.

Graph: Definition, Terminologies, Representations (Set, Linked List, Matrix), Operations: Traversal (BFS, DFS). Applications of Graphs.

Text book:

Classic Data Structure, D. Samanta, PHI, 2/ed.

Reference Books:

- ✓ Ellis Horowitz, Sartaj Sahni, "Fundamentals of Data Structures", Galgotia Pubs.
- ✓ Sastry C.V., Nayak R, Ch. Rajaramesh, Data Structure & Algorithms, I. K. International Publishing House Pvt. Ltd, New Delhi.

Major-1-P -III / Minor-II-P-I: Lab: Data Structures

Write a C Program for the followings

1. To search an element and print the total occurrences in the array.
2. To insert and delete elements into/from appropriate position in an array.
3. To perform Binary Search.
4. To perform Bubble sort.
5. To perform Selection sort.
6. To implement linear linked list and perform operations such as traverse, search, insert, delete, and reversing the list.
7. To implement circular linked list and perform operations such as node insert and delete.
8. To implement double linked list and perform operations such as node insert and delete.
9. To represent a Sparse Matrix using linked list.
10. Polynomial representation using linked list.
11. Array and Linked list implementations of Stack and perform operations such as push, pop and status.

12. Linked list implementation of Queue and perform operations such as enqueue and dequeue.
13. Linked list implementation of Circular Queue.
14. To implement a Binary Search Tree.
15. To perform tree traversal operations.
16. To implement adjacency matrix for a given graph.
17. To perform BFS and DFS traversal.

Major-1-P - IV : Database Management System

Course Objectives:

- To understand the database concepts for efficient storage and retrieval of data.
- To learn about database design and transaction processing

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Build data models using entity relationship concepts
2. Design databases by systematically applying the normalization process
3. Create relational database tables and perform various operations using SQL
4. Learn issues relating to database transactions and approaches to deal with them

Unit-I:

Introduction to Database and Database Users, Database System Concepts and Architecture: data Models, schema, and instances, Conceptual Modeling and Database Design, Entity Relationship (ER) Model: Entity Types, Entity Sets, Attributes, Keys, Relationship Types, Relationship Sets, Roles and Structural Constraints, Weak Entity Types, ER Naming Conventions. Enhanced Entity-Relationship (EER) Model.

Unit-II:

Relational data Model and SQL: Relational Model Concepts, Basic SQLs, SQL Data Definition and Data types, Constraints in SQL, Retrieval Queries in SQL, INSERT, DELETE, UPDATE Statements in SQL, Relational Algebra and Relational Calculus: Unary Relational Operations: SELECT and PROJECT, Binary Relation: JOIN and DIVISION.

Unit-III:

Database Design Theory and Normalization: Functional Dependencies, Normal Forms based on Primary Keys, Second and third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

Unit-IV:

Transaction Processing Concepts: Transaction and System Concepts, Properties of Transactions, Recoverability, Serializability, Concurrency Control Techniques, Locking techniques for Concurrency Control, Concurrency Control based on Time-Stamp Ordering.

Text Books:

- ✓ *Fundamentals of Database Systems, R Elmasri, S B. Navathe, Pearson Education*
- ✓ *Database Management Systems, Rajiv Chopra, S. Chand Pubs.*

Reference Book:

An Introduction to Database System, Date C. J. - Pearson Education, New Delhi.

Core IV-Lab: Data Base Management System

Create and use the following database schema to answer the given queries.

EMPLOYEE Schema

Field	Type	NULL	KEY	DEFAULT
Eno	Char(3)	NO	PRI	NIL
Ename	Varchar(50)	NO		NIL
Job_type	Varchar(50)	NO		NIL
Manager	Char(3)	Yes	FK	NIL
Hire_date	Date	NO		NIL
Dno	Integer	YES	FK	NIL
Commission	Decimal(10,2)	YES		NIL
Salary	Decimal(7,2)	NO		NIL

DEPARTMENT Schema

Field	Type	NULL	KEY	DEFAULT
Dno	Integer	No	PRI	NULL
Dname	Varchar(50)	Yes		NULL
Location	Varchar(50)	Yes		New Delhi

List of Queries:

1. Display Employee Name, Job, Hire Date, Employee Number; for each employee with the Employee Number appearing first.
2. Display unique Jobs from the Employee Table.
3. Display the Employee Name concatenated by a Job separated by a comma.
4. Display all the data from the Employee Table. Separate each Column by a comma and name the said column as THE_OUTPUT.
5. Display the Employee Name and Salary of all the employees earning more than \$2850.
6. Display Employee Name and Department Number for the Employee No= 7900.
7. Display Employee Name and Salary for all employees whose salary is not in the range of \$1500 and \$2850.
8. Display Employee Name and Department No. of all the employees in Dept 10 and Dept 30 in the alphabetical order by name.
9. Display Name and Hire Date of every Employee who was hired in 1981.
10. Display Name and Job of all employees who don't have a current Manager.
11. Display the Name, Salary and Commission for all the employees who earn commission.
12. Sort the data in descending order of Salary and Commission.
13. Display Name of all the employees where the third letter of their name is 'A'.
14. Display Name of all employees either have two 'R's or have two 'A's in their name and are either in Dept No = 30 or their Managers Employee No = 7788.
15. Display Name, Salary and Commission for all employees whose Commission Amount is 14 greater than their Salary increased by 5%.
16. Display the Current Date.
17. Display Name, Hire Date and Salary Review Date which is the 1st Monday after six months of employment.
18. Display Name and calculate the number of months between today and the date each employee was hired.
19. Display the following for each employee <E-Name> earns < Salary> monthly but wants <3*Current Salary>. Label the Column as Dream Salary.
20. Display Name with the 1st letter capitalized and all other letters lower case and length of their name of all the employees whose name starts with 'J', 'A' and 'M'.
21. Display Name, Hire Date and Day of the week on which the employee started.
22. Display Name, Department Name and Department No for all the employees.
23. Display Unique Listing of all Jobs that are in Department # 30.
24. Display Name, Department Name of all employees who have an 'A' in their name.
25. Display Name, Job, Department No. and Department Name for all the employees working at the Dallas location.
26. Display Name and Employee no. Along with their Manager's Name and the Manager's employee no; along with the Employees Name who do not have a Manager.
27. Display Name, Department No. And Salary of any employee whose department no. and salary matches both the department no. And the salary of any employee who earns a commission.
28. Display Name and Salaries represented by asterisks, where each asterisk (*) signifies \$100.
29. Display the Highest, Lowest, Sum and Average Salaries of all the employees.
30. Display the number of employees performing the same Job type functions.
31. Display the no. of managers without listing their names.
32. Display the Department Name, Location Name, No. of Employees and the average salary for all employees in that department.
33. Display Name and Hire Date for all employees in the same dept. as Blake.
34. Display the Employee No. And Name for all employees who earn more than the average salary.
35. Display Employee Number and Name for all employees who work in a department with any employee whose

name contains a 'T'.

36. Display the names and salaries of all employees who report to King.
37. Display the department no, name and job for all employees in the Sales department.

SEMESTER-III

Major-1-P: Discrete Mathematical Structures

Course Outcomes:

- To learn the mathematical foundations required for computer science.
- This course will help in understanding other courses in computer science.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn propositional logic and set theory
2. Learn concept of functions and recurrence relations
3. Learn counting techniques, and relations
4. Learn concepts of graphs and its applications

Unit-I:

Logic and Proofs: Propositional logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs.

Sets: Venn Diagrams, Subsets, The size of a set, Power Sets, Cartesian Products, Set Operations

Unit-II:

Functions: One-to-One and Onto Functions, Inverse Functions and Compositions of Functions Partial Functions. Sequences, Recurrence Relations, Summations.

Unit-III:

Counting: The Basics of Counting, The Pigeonhole Principle, Permutations and Combinations, Binomial Coefficients and Identities.

Relations: Relations and their Properties, n-ary Relations and their Applications, Representing Relations, Closure of Relations, Equivalence Relations, partial Orderings.

Unit-IV:

Graph Terminology and Special Types of Graphs, Bipartite Graphs, Representing Graphs: Isomorphism of Graphs, Euler and Hamilton Paths, Shortest Path Problems: Dijkstra's Algorithm, Traveling Salesperson Problem, Planar Graphs, Graph Coloring. Tree Traversal, Minimum Spanning Trees

Text Books:

- ✓ *Kenneth H. Rosen, Discrete Mathematics and its Applications, Mc Graw Hills International Seventh Edition.*
- ✓ *C. L. Liu, "Elements of Discrete Mathematics", McGraw Hills International Second Edition.*

Reference Books:

- ✓ *Elements of Discrete Mathematics by C. L. Liu and D.P. Mohapatra, TMH, 2012*
- ✓ *J. P Tremblay, R. Manohar, "Discrete Mathematical Structures with Applications to Computer Science", TMH, 1997.*

Major-1-P - VI: Computer Organization

Course Objectives:

- To understand data representation techniques and used of various logic gates
- To understand the basic components of a digital computer and their working
- To know about various memory devices

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Use different number systems and know the function of basic logic gates
2. Design various combinational circuits

3. Understand the functioning of a digital computer
4. Understand the use of various memory devices and their management

Unit-I:

Character Codes, Decimal System, Binary System, Decimal to Binary Conversion, Hexadecimal Notation, Boolean Algebra, Basic Logic Functions: Electronic Logic Gates, Synthesis of Logic Functions, Minimization of Logic Expressions, Minimization using Karnaugh Maps, Synthesis with NAND and NOR Gates, Tri-State Buffers.

Unit-II:

Designing of combinational circuits- Half Adder, Full Adder, Carry-Lookahead Addition, Decoders, Encoders, Multiplexers, Flip-Flops, Gated Latches, Master-Slave Flip-Flops, Edge-Triggering, T Flip-Flops, JK Flip-Flops. Registers and Shift Registers, Counters.

Unit-III:

Basic Structure of Computers - Computer Types, Functional Modules, Basic operational Concepts, Bus Structures, Performance, Multiprocessors and Multi-computers, Input/Output Organization - Accessing I/O devices, Interrupts, Processor examples, Direct memory access, Buses, Interface circuits, Standard I/O interfaces.

Unit-IV:

Memory System - Basic concepts, Semi-conductor RAM memories, Read-only memories, Speed, Size and Cost, Cache Memory: Computer Memory System, Cache Memory Principles, Performance considerations, Virtual Memories, memory management requirements, Secondary Storage.

Text Books:

- ✓ *M. Morris Mano, Michael D. Ciletti (2008), Digital Design, 4th edition, Pearson Education Inc, India.*
- ✓ *Carl Hamacher, Zvonks Vranesic, SafeaZaky (2002), Computer Organization, 5th edition, McGraw Hill, New Delhi, India*

Reference Books:

- ✓ *Computer Architecture and Organization: William Stallings, Pearson Education.*
- ✓ *Computer Architecture and Organization: John P. Hayes McGraw Hill.*
- ✓ *Computer Organization and Design Hardware/ Software Interface: David A. Patterson, John L. Hennessy, Elsevier.*
- ✓ *Computer Architecture & Organization, Rajiv Chopra, S. Chand Pubs.*

Major-1-P - VI-Lab: Computer Organization

Computer Organization Lab Manual

1. To perform conversion between number systems and understand character codes.
2. To verify the truth tables of basic logic gates.
3. To simplify Boolean expressions.
4. To implement logic gates using NAND and NOR gates only.
5. To design and verify Half Adder and Full Adder circuits.
6. To design combinational circuits.
7. To study different types of flip-flops.
T Flip-Flops, JK Flip-Flops etc
8. To study registers and shift registers.
9. To design and verify counters.
10. To understand memory and I/O organization.

Major-1-P / Minor-I-P-II: Operating Systems

Course Objectives:

- To understand Operating system structure and services.
- To understand the concepts of Process, memory, storage, and I/O management.
- To explore different applications of data structures.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Understand various services offered by an OS as a resource manager
2. Understand the concept of a process and various CPU scheduling techniques
3. Learn the concepts on effective memory management and virtual memory
4. Learn various approaches to disk scheduling & file management techniques

Unit-I:

Introduction to Operating System, Computer System Architecture, System Structures: Operating system services, User and Operating-System Interface, system calls, system programs, Operating system design and implementation, Operating system structure, Batch processing, multi-programming, time-sharing and real-time systems

Unit-II:

Process Management: Process Concept, Operations on processes, Process scheduling, Inter- process Communication, Threads, Multithreading Models. CPU Scheduling algorithms: Scheduling Criteria, FCFS, SJF, Priority, Round Robin, Multilevel Queue, Multilevel Feedback Queue. Deadlocks: Deadlock detection, deadlock prevention, and deadlock avoidance fundamentals.

Unit-III:

Memory Management Strategies: Swapping, Contiguous Memory Allocation, Segmentation, Paging, Virtual Memory Management: Concepts, Demand Paging, Page Replacement techniques: FIFO, LRU, Optimal, Thrashing.

Unit-IV:

Storage Management: Overview of Mass-Storage Structure, Disk Scheduling: FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK, RAID technology.

File System concept, Access Methods, Directory and Disk Structure, File System systems, File Sharing and File Protection.

Text Books:

- ✓ *Operating System Concepts, Abraham Silberschatz, Peter B. Galvin, and Greg Gagne, Eighth Edition, Wiley Student Edition 2009*
- ✓ *Operating Systems, Rajiv Chopra, S. Chand Pubs.*

Reference Books:

- ✓ *Modern Operating System, Tanenbaum, Pearson, 4/ed. 2014*
- ✓ *Operating Systems 5th Edition, William Stallings, Pearson Education India*
- ✓ *Richard Blum, Linux Command Line and Shell Scripting Bible, O' Reilly*

Major-1-P / Minor-I-P-II: -Lab: Operating Systems

1. Basic Linux Commands and Overview (date, cal, who, tty, echo, bc, pwd, mkdir, rmdir, cd, cat, cp, mv, rm, ls, wc)
2. Shell Programming
 - i. Write a shell script to perform the tasks of basic calculator.
 - ii. Write a shell script to find the greatest number among the three numbers.
3. Shell Programming
 - i. Write a shell script to check if the number entered at the command line is prime or not.
 - ii. Write a shell script to display the multiplication table of any number.
4. Shell Programming
 - i. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
 - ii. Write a shell script to find the sum of digits of a given number.
 - iii. Write a shell script to find the factorial of a given number.
5. Write a program (using fork() and/or exec() commands) where parent and child execute:
 - i. Same program, same code.
 - ii. Same program, different code.
 - iii. Before terminating, the parent waits for the child to finish its task.
6. Write a program to copy files using system calls.
7. Write a program using C to implement FCFS scheduling algorithm.
8. Write a program using C to implement Round Robin scheduling algorithm.
9. Write a program using C to implement SJF scheduling algorithm.
10. Write a program using C to implement first-fit, best-fit, and worst-fit allocation strategies.

SEMESTER IV

Major-1-P -VIII: Web Technologies

Course Objectives:

- To understand the essentials of Web Technologies.
- To understand frontend (HTML, CSS, JavaScript) and backend Technologies (PHP) for developing Web Sites.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn the basics of Internet protocols and HTML
2. Learn the use of CSS
3. Learn the use of Java Scripts
4. Learn the use of PHP and design a Website

Unit-I:

Introduction to the Internet, Internet Protocols, World Wide Web (WWW): Introduction, History, HTTP and HTTP methods, Web Browser, Web Server and their examples, Web page, working principles of WWW. Web Development: Introduction, Front-end and Backend Development Technologies. Concepts of Client-Server communication.

HTML: Introduction, characteristics, basic structure of an HTML document, understanding basic HTML tags and attributes, creating an HTML document. Working with tags for text- formatting, lists, hyperlink, images, tables, frames, multimedia. HTML forms and its basic elements.

Unit-II:

Cascading Style Sheets (CSS): Introduction, Benefits of using CSS, Understanding the Syntax, CSS Selectors, Using CSS: External, Internal Inline CSS. Comments in CSS. Basic CSS Properties: Color, Background, Text, Font, List, Table, Display.

CSS Box Model: Introduction, working with Margin, Border, and Padding. Pseudo-class & element, working with block elements, Scrolling text, Navigation Bar and Drop Downs.

Unit-III:

JavaScripts: Introduction, Features, Benefits, Creating Simple JavaScript. Using JavaScript in HTML. Exploring Pop-up Boxes: alert, confirm, prompt box. Displaying outputs in JavaScript.

Programming using JavaScript: Data types, Variables, Operators, Expressions (Arithmetic, String, Logical), Comments. Control Statements: Conditional, Looping and Jump Statements. Functions (built-in & user defined) and their usage. Working with Array and Date Objects. Introduction to DOM, Event handling and Form validation in JavaScript.

Unit-IV:

PHP: Features, Print/echo statement, Data Types, Variables, Constants, Strings, Arrays, Operators. Control Structures: Conditional, Looping & Jump Statements. Brief overview of Arrays, Functions: String, Date-Time, Mathematical and User-defined functions.

Embedding PHP in HTML, Reading Form data of a Web Page. Introduction to PHP with Database: Connecting to Database, Creating Table, inserting records, modifying data and retrieving data and displaying in HTML.

Text Book:

Web Technologies (Black Book), DreamTech Press

Reference Books:

- ✓ *Web Enabled Commercial Application Development Using HTML, JavaScript, DHTML and PHP 4th Edition by Ivan Bayross.*
- ✓ *HTML, XHTML and CSS Bible, 5ed, Wiley India- Steven M. Schafer.*

Major-1-P -VIII-Lab: Web Technologies

1. Create a Web Page to display "Hello HTML".
 - ☐ Display the same using different headings: h1 to h6
 - ☐ Apply bold, italic formats
 - ☐ Change text color, background of the page
2. Create a Web Page to display the list of Fruits using both ordered and unordered list. Recreate the lists using images of the same items.
3. Use the web page created in Question Number 2. When the user clicks on the image of a Fruit, it should open a new page that contains an image of the Fruit along with its benefits (use multiple paragraphs, and make sure the image is aligned properly).
4. Create a web page that displays details of the Fruits in Tabular format. Use serial no., name, color, taste, price/kg.

5. Create a Web Page that displays a video file (record/create a video of your own).
 - ☐ Apply various controls such as play, pause, volume.
 - ☐ Apply autoplay, muted and both.
 - ☐ Display a Youtube Video in your Web Page.
6. Design a Student registration form to collect various data about a Student which includes Name, Age, Gender (M, F, O), Mobile No., Email ID, Stream (Science/Arts/Commerce indrop-drown), Choice for participating in NCC, NSS, YRC (use check box), and two buttons for reset and submit respectively. Display the form at the center of the page with proper alignment of each item in the form.
7. Use the web page created in Question no. 4. Use CSS and apply various styling to the text, colors to each row of the table, styling to borders and background color of the table.
8. Create a Web page with a Horizontal Navigation bar containing four items such as Home, College, Students, Teachers. The first item should be active, by default. The background color of the item changes when the user moves the mouse over it.
9. Create a Web Page that asks the user to enter the number of Students, and then iteratively ask the details of each Student and display them in a list, using JavaScript.
10. Modify the above program (Q. 9) to display the details in a tabular format, dynamically using JavaScript.
11. Create a Web Page with two text fields and four buttons to perform arithmetic operations such as Addition, Subtraction, Multiplication and Division. The user has to enter numbers in the text fields and press any of the above buttons. The JavaScript program should perform corresponding operations and display the result in the same page. [NOTE: Use different functions for each operation]
12. Use JavaScript and validate the form data in the Student registration Page (created in Q. no. 6). When the user clicks the Submit button, the program has to validate that every field contains valid data. [NOTE: validate name, age, gender, email, mobile number]
13. Write a PHP program that asks the user to enter a number and finds the factorial of it.
14. Write a PHP program that creates a Table in a database with a number of columns as determined by the fields in the Student registration form created above.
15. Write a PHP program to store the data of the Student registration form in a Database.
16. Write a PHP program that asks the user to enter a Student name and display the Student details retrieved from the database in the same page.
17. Students are required to combine the Student Registration form, and PHP program (s) [which interact with Database] to see the dynamic updation of the Student registration data in the Database when a new Student is registered.

Major-1-P- IX: Object Oriented Programming using C++

Course Objectives:

- To know about the Object-Oriented Programming concepts.
- To write object-oriented programs using C++ constructs

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Understand OOPs concepts as a programming style
2. Use class/objects in programs and functions of different types
3. Learn the concept of inheritance and overloading of functions and operators
4. Use files in C++

Unit-I:

Principles of Object-Oriented Programming: Object-Oriented Programming (OOP) Paradigm, Basic Concepts of OOP, Benefits of OOP, Characteristics of OOPS, Object Oriented Languages, Applications of OOP.

Introduction to C++, Difference between C & C++, Tokens, Data types, Operators, Structure of C++ Program, C++ statements, Expressions and Control Structures.

Functions in C++: Argument passing in function, Inline Functions, Default Arguments, Const. Arguments, Friend function.

Unit-II:

Classes and Objects: Defining Member Functions, Making an outside Function Inline, Nested Member Functions, Private Member Functions, Arrays within a Class, Memory Allocation for Objects, Static Data Members, Static Member Functions, Arrays of Objects, Objects as Function Arguments, Friend Functions.

Constructors & Destructors: Constructors, Parameterized Constructors, Constructors with Default Arguments, Dynamic

Initialization of Objects, Copy Constructor, DynamicConstructors, Destructors.

Unit-III:

Inheritance: Basics of Inheritance, Type of Inheritance, Virtual Base Classes, AbstractClasses, Member Classes, Nesting of Classes. Polymorphism: Pointers, Pointers to Objects, this Pointer, Pointers to Derived Classes, Virtual Functions, Pure Virtual Functions, Function Overloading, Operator Overloading.

Unit-IV:

Managing Console I/O Operations: C++ Streams, C++ Stream Classes, Unformatted I/O Operations, Formatted Console I/O Operations, Managing Output with Manipulators.

Files: Classes for File Stream Operations, Opening and Closing a File, Detecting end-of-file, File Modes, File Pointers and their Manipulations, Sequential Input and Output Operations, Updating a File: Random Access, Error Handling during File Operations, Command-line Arguments.

Text Books:

- ✓ E. Balgurusawmy, *Object Oriented Programming with C++, 4/e (TMH)*.
- ✓ Paul Deitel, Harvey Deitel, *"C++: How to Program", 9/e. Prentice Hall*.

Reference Books:

- ✓ Bjarne Stroustrup, *Programming - Principles and Practice using C++, 2/e, Addison-Wesley*
- ✓ Herbtz Schildt, *C++: The Complete reference, McGrawHill*.

Major-1-P- IX-Lab: Programming using C++

1. Write a Program for Swapping of two numbers.
2. Write a Program to find sum of four numbers using default argument passing.
3. Write a Program to find square and cube of a number using inline function.
4. Write a Program to find the factorial of a number.
5. Write a Program to find reverse of a number.
6. Write a program to find sum of four numbers using defaultargument passing inmember function.
7. Write a Program to find area of circle, triangle and rectangle using functionoverloading.
8. Write a program to distinguish the properties of static and non-static ata members.
9. Write a program to show the method of accessing static private member function.
10. Write a program to show the ways of calling constructors and destructors.
11. Write a program to perform ++ operator overloading using member function.
12. Write a program to perform ++ operator overloading using friend function.
13. Write a program to perform + operator overloading for two complex numberaddition.
14. Write a program to perform + operator overloading for string concatenation.
15. Write a program to perform single inheritance.
16. Write a program to perform multiple inheritance.
17. Write a program to create an integer array using new operator and findthe sum andaverage of array elements.
18. Write a program to implement virtual destructor.
19. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).
20. Write a program to Copy the contents of one file to other.

Major-1-P- X / Minor-II-P-II: Computer Network

Course Objectives:

- To understand data communication and network concepts.
- To learn about different communication standards
- To understand different network protocols

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Understand concepts on data communication and the use of communication devices
2. Learn about analog and digital signals and basic components of data communication
3. Learn about errors during data communication & access control mechanisms
4. Learn various network protocols and network security issues

Unit-I:

Introduction to Data Communications and Network Models: Protocols and Standards, Layers in OSI Models, Analog and Digital Signals, Transmission Modes, Transmission Impairment, Data Rate Limits, Performance, Digital Transmission, Network Devices & Drivers: Router, Modem, Repeater, Hub, Switch, Bridge (fundamental concepts only).

Unit-II:

Signal Conversion: Digital-to-Digital Conversion, Analog-to-Digital Conversion, Digital-to-analog Conversion, Analog-to-Analog Conversion. Transmission Media: Guided Media, Unguided Media, Switching Techniques: Packet Switching, Circuit Switching, Datagram Networks, Virtual-Circuit Networks, and Structure of a Switch.

Unit-III:

Error Detection and Correction: Checksum, CRC, Data Link Control: Framing, Flow and Error Control, Noiseless Channels, Noisy channels, (Stop and Wait ARQ, Sliding Window Protocol, Go Back N, Selective Repeat) HDLC, Point-to-Point Protocol. Access Control: TDM, CSMA/CD, and Channelization (FDMA, TDMA, and CDMA).

Unit-IV:

Network Layer: Logical Addressing, IPv4 Addresses, IPv6 Addresses, Virtual-Circuit Networks: Frame Relay and ATM, Transport Layer: Process-to-Process Delivery: UDP, TCP. Application layers: DNS, SMTP, POP, FTP, HTTP, Basics of WiFi (Fundamental concepts only), Network Security: Authentication, Basics of Public Key and Private Key, Digital Signatures and Certificates (Fundamental concepts only).

Text Book:

Data Communications and Networking, Fourth Edition by Behrouza A. Forouzan, TMH.

Reference Book:

Computer Networks, A. S. Tanenbaum, 4th edition, Pearson Education.

Major-1-P- X / Minor-II - Lab: Computer Network

1. Use the **ipconfig** (Windows) or **ifconfig** (Linux/Mac) command to display the current network configuration.
 - i. Identify and document the IP address, subnet mask, and default gateway of the system.
 - ii. Change the IP address of the system using **netsh** (Windows) or **ifconfig** (Linux/Mac). Verify the change using the same command.
 - iii. Experiment by configuring static IP, dynamic IP.
2. Use the **ping** command
 - i. to check connectivity between Systems in your Lab.
 - ii. to a remote server (e.g., google.com).
 - iii. Analyze the round-trip time and packet loss.
3. Use the **tracert** (Windows) or **traceroute** (Linux/Mac) command to trace the path to a remote server. Document the intermediate hops and their IP addresses.
4. Use the **netstat** command to display active connections, listening ports, and network statistics.
 - i. Document and explain the various parameters and their significance.
 - ii. Use **netstat -r** or **route** to display the routing table of your system. Identify the default gateway and other routes.
5. Use the **arp -a** command to display the ARP table of your system.
 - i. Identify the MAC addresses corresponding to different IP addresses.
 - ii. Clear the ARP cache using **arp -d** and verify the cache is cleared. Re-populate the ARP table by pinging different hosts on the network and verify the entries.
6. Use the **nslookup** command to query the DNS records of a domain (e.g., google.com).
 - i. Identify and document the IP addresses associated with the domain.
 - ii. Use the **dig** command (Linux/Mac) for a more detailed DNS query and compare the output with **nslookup**.
7. Use the **nmcli** command (Linux) or **netsh wlan show networks** (On Windows) to scan for available Wi-Fi networks and connect to a specified network. Document the steps and verify the connection.
8. Use the **tcpdump** command (Linux) or **Wireshark** to capture network packets.

- i. Capture and analyze traffic for a specific protocol (e.g., HTTP) and identify key details like source and destination IPs, ports, and packet content.
 - ii. Filter captured packets to display only traffic to/from a specific IP address or port using **tcpdump** for Wireshark filters.
- 9. Use the **nmap** command to perform a network scan of your local network.
 - i. Identify active hosts, open ports, and running services.
 - ii. Perform a more detailed scan with service/version detection using **nmap -sV** and analyze the results.
- 10. Use the **iptables** command (Linux) to set up basic firewall rules. On Windows, use **netsh advfirewall**. Block all incoming traffic except for SSH and HTTP, and verify the rules are working.
- 11. Use the **route** command to add a static route to a specific network.
 - i. Verify the route using **route -n** (Linux) or **route print** (Windows).
 - ii. Set up IP forwarding on a Linux system using **sysctl** to enable packet forwarding. Test the configuration by pinging through the system acting as a router.

SEMESTER V

Major-1-P -XI: Software Engineering

Course Objectives:

- To understand importance of Software engineering.
- To understand different software development models
- To understand various issues involved in a software development project

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Understand various software development lifecycle models
2. Know the complexities involved in software development projects & how to deal with them
3. Understand the software design process starting from requirement analysis
4. Learn about software documentation, software testing and maintenance

Unit-I:

Introduction: Evolution of Software to an Engineering Discipline, Software Development Projects, Exploratory Style of Software Development, Emergence of Software Engineering, Changes in Software Development Practices, Computer Systems Engineering. Software Lifecycle Models: Waterfall Model and its Extensions, Rapid Application Development (RAD), Agile Development Models, Spiral Model.

Unit-II:

Software Project Management: Software Project Management Complexities, Responsibilities of a Software Project Manager, Project Planning, Metrics for Project Size Estimation, Project Estimation Techniques, Empirical Estimation Techniques, COCOMO, Halstead's Software Science, Staffing Level Estimation, Scheduling, Organization and Team Structures, Staffing, Risk Management, Software Configuration Management.

Unit-III:

Requirement Analysis and Specification: Requirements Gathering and Analysis, Software Requirement Specifications, Formal System Specification Axiomatic Specification, Algebraic Specification, Executable Specification and 4GL.

Software Design: Design Process, characterize a Good Software Design, Cohesion and Coupling, Layered Arrangements of Modules, Approaches to Software Design (Function Oriented & Object-Oriented).

Unit-IV:

Coding and Testing: Coding: Code Review, Software Documentation, Testing, Unit Testing, Black Box and White Box Testing, Debugging, Program Analysis Tools, Integration Testing, System Testing, Software Maintenance.

Text Book:

Fundamental of Software Engineering, Rajib Mall, Fifth Edition, PHI Publication, India.

Reference Books:

- ✓ *Software Engineering– Ian Sommerville, 10/Ed, Pearson.*
- ✓ *Software Engineering Concepts and Practice – Ugrasen Suman, CengageLearning India Pvt, Ltd.*
- ✓ *Software Engineering, R Khurana, Vikash Pubs.*

Major-1-P -XI: Software Engineering

Students have to do at least two software development projects from the list of projects given below. They have to follow the complete software development lifecycle with the following details. UML can be used as a design tool. (Coding is optional).

1.
 - Problem Statement
 - Process Model
2. Requirement Analysis:
 - Creating a Data Flow
 - Data Dictionary, Use Cases
3. Project Management:
 - Computing FP
 - Effort
 - Schedule, Risk Table, Timeline chart
4. Design Engineering:
 - Architectural Design
 - Data Design, Component Level Design
5. Testing:

- Basis Path Testing

List of Projects:

1. Criminal Record Management: Implement a criminal record management system for jailers, police officers and CBI officers.
2. Route Information: Online information about the bus routes and their frequency and fares
3. Car Pooling: To maintain a web-based intranet application that enables the corporate employees within an organization to avail the facility of carpooling effectively.
4. Patient Appointment and Prescription Management System
5. Organized Retail Shopping Management Software
6. Online Hotel Reservation Service System
7. Examination and Result computation system
8. Automatic Internal Assessment System
9. Parking Allocation System
10. Wholesale Management System

Major-1-P - XII(A) / Minor-I-P-III : Introduction to Artificial Intelligence

Course Objectives:

- To learn the basic concepts of AI.
- To understand AI problem-solving approaches

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Understand state space search as an approach to AI problem solving
2. Understand various Knowledge Representation techniques
3. Learn the complexity involved in NLP & role of learning in AI problem-solving
4. Understand the importance of Expert systems and the use of AI programming languages.

Unit-I:

Introduction to AI, Scope of AI, Characteristics of AI problems, Turing test, Concept of Intelligent agents, Approaches to AI problem-solving, State space search, production system, Uninformed search: Breadth-First, Depth-First, Iterative deepening, bidirectional and beam search.

UNIT-2:

Informed/Heuristic search: Generate-and-Test, Hill climbing, Best-first search, A* algorithm, Problem reduction, AO*, Constraint satisfaction, Solution of CSP using search, Means-End analysis.

UNIT-3:

Knowledge Representation: Propositional logic and Predicate logic along with their resolution principles, Unification algorithm, forward and backward chaining and conflict resolution, Semantic nets, Frames, Conceptual dependencies, Scripts.

Reasoning under uncertainty: Bayesian Belief networks, Dempster Shafer theory

UNIT-4:

Natural language processing: Introduction, Levels of knowledge in language understanding, , Phases of Natural language understanding, top-down and bottom-up parsing, transition networks.

Expert Systems: Introduction, Architecture, Expert system development cycle, Examples of ES: Mycin and Dendral.

Text Books:

- *Artificial Intelligence - A Modern Approach* by Stuart J. Russell & Peter Norvig, Prentice Hall
- *Artificial Intelligence* by Rajiv Chopra, S. Chand Pubs.

Reference Books:

- *D.W. Patterson, Introduction to A.I and Expert Systems*, PHI Pub.
- *Artificial Intelligence* by Rich, Knight, and Nair, McGraw Hill

Major-1-P - XII(A)-Lab: Artificial Intelligence

1. Write a Python program to implement Depth-First Search (DFS) for a given graph. Test your program on a graph with at least 5 nodes. Verify your program by printing the order in which nodes are visited.
2. Write a Python program to implement Breadth-First Search (BFS) for a given graph. Use a queue to manage the nodes to be explored. Test your program on a graph with at least 5 nodes and print the order of node visits
3. Write a Python program to implement Uniform Cost Search (UCS) for finding the shortest path in a weighted graph. Test your program on a graph with at least 5 nodes and varying edge weights.

4. Write a Python program to implement the A* search algorithm. Your program should take a graph, a start node, a goal node, and a heuristic program as input. Test your implementation on a grid-based graph where the heuristic is the Manhattan distance.
5. Write a Python program to implement Greedy Best-First Search. Use a heuristic program to guide the search.
6. Write a Python program to solve a maze using the A* search algorithm. Represent the maze as a grid, where 0 indicates an open cell and 1 indicates a wall. Use Manhattan distance as the heuristic.
7. Write a Python program to implement the Minimax algorithm with Alpha-Beta pruning for a simple game (e.g., Tic-Tac-Toe).
8. Write a Python program to implement the Hill Climbing algorithm with random restarts. Test your program on a problem where the solution landscape has multiple peaks.
9. Write a Python program to represent the state of the 8-puzzle. Use a 2D list or a singlelist with 9 elements to represent the tiles. Implement a program to display the puzzle state.
10. Write a Python program to generate all possible moves (up, down, left, right) from a given state in the 8-puzzle. Ensure that your program checks for the boundaries of the puzzle.

Major-1-P - XII(B) : Computer Graphics

Course Objectives:

- To understand basic concepts of computer graphics.
- To learn techniques for creating basic graphical structures
- To learn different transformation techniques

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Know the use of different graphics systems
2. Learn different algorithms to draw geometrical figures
3. Learn various geometric transformation techniques
4. Learn techniques for clipping

Unit-I:

Computer Graphics: A Survey of Computer graphics, Overview of Graphics System: Video Display Devices, Raster-Scan Systems, Input Devices, Hard-Copy Devices, Graphics Software.

Unit-II:

Graphics Output Primitives: Point and Lines, Algorithms for line, circle & ellipse generation, Filled-Area Primitives. Attributes of Graphics Primitives: Point, line, curve attributes, fillarea attributes, Fill methods for areas with irregular boundaries.

Unit-III:

Geometric Transformations (both 2-D & 3-D): Basic Geometric Transformations, Transformation Matrix, Types of transformation in 2-D and 3-D Graphics: Scaling, Reflection, shear transformation, rotation, translation. 2-D, 3-D transformation using homogeneous coordinates.

Unit-IV:

Two-Dimensional Viewing: Introduction to viewing and clipping, viewing transformation in 2-D, viewing pipeline, Clipping Window, Clipping Algorithms: Point clipping, Line clipping and Polygon clipping.

Text Books:

- *Donald Hearn & M. Pauline Baker, "Computer Graphics with OpenGL", Pearson Education.*
- *Mathematical Elements for Computer Graphics, D. F. Rogers & J. A. Adams, MGH, 2nd ed.*

Reference Books:

- *Computer Graphics principles & practice, Foley, Van Dam, Feiner, Hughes Pearson Education*
- *Computer Graphics by Zhigang Xiang, Roy A Plastic, McGraw-Hill*

Major-1-P - XII(B)-Lab: Computer Graphics using OpenGL

1. Write a program to implement Bresenham's line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to fill a polygon using Scan line fill algorithm.
6. Write a program to apply various 2D translation transformation.
7. Write a program to apply 2D object homogenous coordinates translation.
8. Write a program to apply various 2D rotation transformation.

9. Write a program to apply 2D object homogenous coordinates rotation.
10. Write a program to apply various 2D scaling transformation.
11. Write a program to apply 2D object homogenous coordinates scaling transformation.
12. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.

Major-1-P -XIII: Programming in Java

Course Objectives:

- To learn Java for writing object-oriented programs
- To understand the use of different Java programming constructs
- To learn exception handling in Java and use of threads.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn the basics of Java programming
2. Create classes/objects and implement different forms of inheritance
3. Use arrays and files in Java
4. Learn about exception handling

Unit-I:

Introduction to Java: Java History, Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords (super, this, final, abstract, static, extends, implements, interface), Data Types, Wrapper class, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type Checking, Built-in Java Class Methods). Input through keyboard using Command line Argument, the Scanner class, BufferedReader class.

Unit-II:

Object-Oriented Programming Overview: Principles of Object-Oriented Programming, Defining & Using Classes, Class Variables & Methods, Objects, Object reference, Objects as parameters, final classes, Garbage Collection. Constructor-types of constructors, this keyword, super keyword. Method overloading and Constructor overloading. Aggregation vs Inheritance, Inheritance: extends vs implements, types of Inheritance, Interface, Up-Casting, Down-Casting, Auto-Boxing, Enumerations, Polymorphism, Method Overriding and restrictions. Package: Pre-defined packages and Custom packages.

Unit-III:

Arrays: Creating & Using Arrays (1D, 2D, 3D and Jagged Array), Array of Object, Referencing Arrays Dynamically. Strings and I/O: Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, StringBuffer Classes and StringBuilder Classes. IO package: Understanding StreamsFile class and its methods, Creating, Reading, Writing using classes: Byte and Character streams, FileOutputStream, FileInputStream, FileWriter, FileReader, InputStreamReader, PrintStream, PrintWriter. Compressing and Uncompressing File.

Unit-IV:

Exception Handling, Threading, Networking and Database Connectivity: Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

Text Book:

E. Balagurusamy, “Programming with Java”, TMH, 4/Ed

Reference Book:

Herbert Schildt, “The Complete Reference to Java”, TMH, 10/Ed.

Major-1-P -XIII Lab: Programming in Java

1. To find the sum of any number of integers entered as command line arguments.
2. To find the factorial of a given number.
3. To convert a decimal to binary number.
4. To check if a number is prime or not, by taking the number as input from the keyboard.
5. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument.

6. Write a program that show working of different functions of String and StringBuffer class like setCharAt(), setLength(), append(), insert(), concat() and equals().
7. Write a program to create a – “distance” class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer
8. Modify the – “distance” class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
9. Write a program to show that during function overloading, if no matching argument is found, then Java will apply automatic type conversions (from lower to higher data type).
10. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword.
11. Write a program to show the use of static functions and to pass variable length arguments in a function.
12. Write a program to demonstrate the concept of boxing and unboxing.
13. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
14. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate Fibonacci series is given in a different file belonging to the same package.
15. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
16. Write a program – “DivideByZero” that takes two numbers a and b as input, computes a/b, and invokes Arithmetic Exception to generate a message when the denominator is zero.
17. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
18. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Throwable).
19. Write a program to demonstrate priorities among multiple threads.
20. Write a program to demonstrate different mouse handling events like mouseClicked(), mouseEntered(), mouseExited(), mousePressed(), mouseReleased() & mouseDragged().
21. Write a program to demonstrate different keyboard handling events.

SEMESTER - VI

Major-1-P - XIV: Algorithm Design Techniques

Course Objectives:

- To understand the importance of algorithm design.
- To learn ways to analyze algorithms
- To learn about adoption of different algorithmic styles for solving different types of problems

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn approaches to algorithm analysis & design
2. Learn different searching and sorting techniques
3. Learn greedy techniques for problem-solving
4. Learn graph-based techniques for practical problem-solving

Unit-I:

Algorithm specification: Pseudo code, Asymptomatic Analysis, Space complexity and time complexity, Analysis and design of Insertion sort algorithm, Divide and Conquer paradigm, Recurrence relations, Solving Recurrences: Substitution methods, Recursion tree method, and Master method.

Unit-II:

Searching and Sorting: Analysis of Linear Search, Binary Search, Merge Sort and Quick Sort, Heap Sort. Hashing: Hash functions, Hash table, Collision resolution: Chaining and Open Addressing (Linear probing, Quadratic probing, Double hashing).

Unit-III:

Greedy Technique: General Method, Applications: Fractional Knapsack Problem, Job Sequencing with Deadlines, Huffman Codes.

Dynamic Programming: General Method, Applications: Matrix Chain Multiplication, longest common subsequence, 0/1 Knapsack.

Unit-IV:

Graph Algorithms, Topological sort, Minimum Spanning Trees: Prim's and Kruskal's algorithm, Single-source shortest paths: Bellman-Ford algorithm, Dijkstra's algorithm.

Text Book:

- *Introduction to Algorithms*, by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, PHI.
- *Fundamentals of Computer Algorithms*. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, University Press

Reference Book:

Algorithm Design, by Jon Kleinberg, Eva Tardos.

Core XIV Lab: Algorithm Design Techniques

1. Write C / C++ Program to implement Insertion Sort
2. Write C / C++ Program to implement Merge Sort
3. Write C / C++ Program to implement Quick Sort
4. Write C / C++ Program to implement Heap Sort
5. Write C / C++ Program to implement Hashing
6. Write C / C++ Program to implement Fractional Knapsack
7. Write C / C++ Program to implement Matrix Chain Multiplication
8. Write C / C++ Program to implement Longest Common Subsequence
9. Write C / C++ Program to implement Prim's Algorithm
10. Write C / C++ Program to implement Kruskal's Algorithm
11. Write C / C++ Program to implement Dijkstra's Algorithm

Major-1-P - XV: Project Work-I

A student has to do a Project work under the guidance of a faculty member. After completing the project, the student has to submit a project report which has to be evaluated by an external examiner. The model template for the project report can be as follows:

1. Title of the project
2. Declaration (by the student)

3. Certificate (of the project guide)

4. Acknowledgement

5. Abstract

[Provide a brief summary of your project, including its objectives, methods, and key findings.]

6. Table of Contents

Introduction Literature

Review Methodology

Results Discussion

Conclusion References

7. Introduction

[Describe the background and context of your project, including the problem statement and objectives.]

8. Literature Review

[Review relevant literature related to your project, discussing previous research, theories, and concepts.]

9. Methodology/

[Explain the methods you used to conduct your research or project, including data collection, analysis techniques, and any tools or software used.]

10. Implementation/Software development

11. Results

[Present the findings of your research or project, using tables, figures, or graphs as needed to illustrate key points.]

12. Discussion

[Interpret your results and discuss their implications, relating them back to your research objectives and the broader context of your field.]

13. Conclusion

[Summarize the main findings of your project and their significance, as well as any recommendations for future research or applications.]

14. References

[List all sources cited in your project using a consistent citation style (e.g., APA, MLA).]

The evaluation pattern of the project will be as follows:

i. Problem formulation and definition

ii. Execution of code & results

iii. Documentation

iv. Clarity in presentation

v. Performance in the Viva voce

SEMESTER-VII

Major-1-P - XVI: Theory of Computation

Course Objectives :

To give an overview of the theoretical foundations of computer science from the perspective of formal languages. To illustrate finite state machines to solve problems in computing. To familiarize Regular grammars, context free grammar.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn the use of Deterministic and Nondeterministic Finite Automata
2. Understand the use of Regular Expressions and regular grammars
3. Understand the significance of Context Free Grammars and Push Down Automata
4. Understand the significance of Turing Machine

Unit-I:

Introduction to Finite Automata, Alphabets, Strings, Languages and Problems. FINITE AUTOMATA (FA): Deterministic Finite Automata (DFA)-Formal definition, language of a DFA. Nondeterministic Finite Automata (NFA)-Definition of NFA, the extended transition function, the language of an NFA, Equivalence of NFA and DFA, NFA with Epsilon Transitions, Eliminating Epsilon transitions, Minimization of DFA's.

Unit-II:

REGULAR EXPRESSIONS (RE): Definition of RE, The operators of Regular Expressions, Building RE, Finite Automata and Regular Expressions- Converting DFA's to Regular Expressions, Converting Regular Expressions to Automata, Regular grammars, and FA, proving languages to be non-regular -Pumping lemma, applications, Closure properties of regular languages. Decision properties of Regular Languages.

Unit-III:

Context Free Grammars and Languages: Context Free Grammars (CFG): Definition, Derivations using a grammar, trees, Leftmost and Rightmost derivations. Ambiguity in grammars and languages: Removing ambiguity from grammars, Inherent ambiguity. Properties of Context-Free Languages: Normal forms for CFGs; Eliminating useless symbols, Eliminating epsilon productions, Eliminating unit productions, Chomsky Normal Form (CNF), Pumping Lemma for Context Free Languages. Decision properties of CFL's.

Unit-IV:

Pushdown Automata: Definition, Instantaneous Descriptions of a PDA, The language of a PDA: Acceptance by Final State, Acceptance by empty stack. Equivalence of PDA's and CFG's. Introduction to Turing Machine: Notation, Instantaneous Descriptions for Turing machines, Transition Diagrams, Language, Turing machines and Halting. Universal Turing Machines.

Text Book:

John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, *Introduction to Automata Theory Languages and Computation*, Pearson Education, India.

Reference Books:

- Sipser, M., *Introduction to the Theory of Computation*, 3rd Edition, Cengage Learning
- John C Martin, *Introduction to languages and the Theory of Computation*, McGraw-Hill Publishing.
- *Automata Theory* by K.L.P. Mishra and N. Chandrasekaran, PHI Learning Pvt. Ltd

Major-1-P - XVII: Computer Architecture

Minor-1-P-IV(With Research)

Course Objectives:

- To understand basic concepts of computer architecture
- To understand the functioning of important components
- To learn about parallel architectures

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn how arithmetic operations are carried out
2. Understand the instruction execution cycle
3. Understand pipelining of operations
4. Have an exposure to parallel processing

Unit-I:

Machine Instructions and Programs: Numbers, Arithmetic Operations, and Characters, Memory Locations and

Addresses, Memory Operations, Instruction and Instruction Sequencing, Addressing Modes, Assembly Language. Arithmetic: Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-Operand Multiplication, Fast Multiplication, Integer Division, Floating Point Numbers and Operations.

Unit-II:

Computer Function and Interconnection: Computer Components, Computer Function, Interconnection Structures, Bus Interconnection, PCI. Cache Memory: Computer Memory System, Cache Memory Principles, Elements of Cache Design, Pentium-4 Cache Organization.

Unit-III:

External Memory: Magnetic Disk, RAID, Optical Memory, Magnetic Tape, External Devices, I/O Module, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access, I/O Channels and Processors, FireWire and InfiniBand. CPU Structure and Function: Processor Organization, Register Organization, Instruction Cycle, Instruction Pipelining, The Pentium Processor.

Unit-IV:

Reduced Instruction Set Computer (RISC): Instruction Execution Characteristics, Use of a Large Register File, Compiler-Based Register Optimization, Reduced Instruction Set Architecture, RISC Pipelining
Parallel Organization: Multiple Processor Organizations, Symmetric Multiprocessors, Cache Coherence and MESI Protocol, Clusters, Non-Uniform Memory Access (NUMA), Vector Computation.

Text Books:

- Carl Hamacher, Zvonks Vranesic, SafeaZaky (2002), *Computer Organization*, 5th edition, McGraw Hill, New Delhi, India
- Mano. M. M.: *Computer System Architecture* 3/ed. (PHI)

Reference Books:

- Hayes, J.P.: *Computer Architecture and Organization* 3/ed. (MGH)
- Stallings, W. *Computer Organization and Architecture* 4/ed. (PHI)

Major-1-P - XVII: computer Architecture

1. Representation of numbers and arithmetic operations on signed numbers
2. Design and implementation of fast adders (Ripple Carry and Carry Look-Ahead Adder)
3. Binary multiplication and integer division algorithms
4. Floating point number representation and operations
5. Instruction sequencing and addressing modes (basic assembly simulation)
6. Cache memory simulation and performance analysis
7. Computer interconnection structures and bus system (PCI concept)
8. Programmed I/O, interrupt-driven I/O, and DMA simulation
9. External memory systems (Magnetic disk, RAID, optical storage)
10. RISC architecture, instruction pipelining, and parallel processing concepts

Major-1-P -XVIII: Mathematics for Machine Learning

Minor-1-P-V(With Research)

Course Objectives:

- To learn the mathematical foundations required for Machine Learning.
- With this course, one can apply the mathematical tools of Machine Learning in real life problems.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Understand the algebra and matrix theory.
2. Apply the idea of calculus.
3. Learn foundations of probability.
4. Apply statistical methods in Machine Learning problems.

Unit-I:

Introduction, Vector Spaces, Linear Independence, Dimensions and Basis, Linear Transformations, Norms and Spaces, Orthogonal Complement and projection, Systems of Linear equations- Gaussian Elimination, LU decomposition

Unit-II:

Matrix and Operations: Addition, Multiplication, Transpose, Inverse, Rank of a matrix, Matrix factorization: Matrix properties and trace, Eigenvalues and Eigenvectors, Cholesky decomposition, Singular value decomposition, diagonalization

Gradient, convex sets and convex functions, Properties of convex functions, Gradient of matrices, Numerical Optimization in ML, Gradient descent optimization

Unit-III:

Applications of Probability, Events and Sample Spaces, Axioms of Probability, Conditional Probability, Baye's Theorem, Random Variable, Expectation, Variance, Concept of Distributions – Bernoulli, Binomial, Poisson, Exponential and Normal.

Unit-IV:

Mean, Median, Mode, Standard Deviation, Regression - Linear, Multiple and Logistic regression, Least Square Method.

Text Book:

M. P. Deisenorth, A Aldo Faisal, Cheng Soon Ong, "Mathematics for Machine Learning", Cambridge University Press.

Reference Books:

- Samuel Hack, "Machine Learning Mathematics".
- Richard Han, "Math for Machine Learning".

Major-1-P -XVIII Lab : Mathematics for Machine Learning (Python/MATLAB)

1. Vector spaces, linear independence, basis and dimension
2. Solution of systems of linear equations using Gaussian elimination and LU decomposition
3. Matrix operations: addition, multiplication, transpose, inverse, rank and trace
4. Eigenvalues, eigenvectors, matrix diagonalization, SVD and Cholesky decomposition
5. Gradient computation and gradient descent optimization
6. Convex sets, convex functions and properties (visualization and analysis)
7. Probability basics: sample space, events and conditional probability
8. Bayes' theorem and random variables with expectation and variance
9. Probability distributions: Bernoulli, Binomial, Poisson, Exponential and Normal
10. Statistical analysis and regression: Mean, Median, Mode, Standard Deviation, Linear, Multiple and Logistic Regression using least squares method

Major-1-P - XIX(A): Data Mining

Course Objectives:

- To learn the basic concepts of data mining and data warehouse.
- To know about the use of data mining techniques in solving various real-world problems
- To learn the fundamental techniques of data mining.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Gain knowledge on data mining and OLAP and OLTP operations
2. Learn the techniques of mining frequent itemsets
3. Know about various classification techniques and metrics for their performance evaluation
4. Learn various clustering methods

Unit-I:

Introduction to Data Mining, Major Issues in Data Mining, Types of Attributes: Nominal, Binary, Ordinal, Numeric (Interval-Scaled, Ratio-Scaled), Discrete and Continuous Attributes Data Preprocessing: Need of Data Preprocessing, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization Data Warehouse and OLAP: Data Warehouse, OLTP vs OLAP, Three-tier Data Warehouse Architecture, Data Cube, Schemas for Multidimensional Data Models (Star, Snowflake, Fact Constellation), OLAP Operations

Unit-II:

Frequent Itemsets, Closed Itemsets, Association Rules, Frequent Itemset Mining Methods: Apriori Algorithm, Generating Association Rules from Frequent Itemsets, Improving the Efficiency of Apriori, A Pattern Growth Approach for Mining Frequent Itemsets (FP- growth), Mining Frequent Itemsets using the Vertical Data Format, Mining Multilevel Associations, Mining Multidimensional Associations, From Association Analysis to Correlation Analysis

Unit-III:

Classification: Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Metrics for Evaluating Classifier Performance, Bayesian Belief Networks, Classification by Backpropagation, Support Vector Machine, Genetic Algorithm, Fuzzy Set Approach

Unit-IV:

Clustering: Cluster Analysis, Partitioning Methods (k-Means, k-Medoids), Hierarchical Methods (Agglomerative vs Divisive Hierarchical Clustering, BIRCH, Chameleon), Density-based Methods (DBSCAN, OPTICS, DENCLUE),

Grid-based Methods (STING, CLIQUE) **Text Book:**

- ✓ *Data Mining Concepts and Techniques by J. Han, M. Kamber and J. Pei, Elsevier.*
- ✓ *Introduction to Data Mining by Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, Pearson Education.*

Major-1-P - XIX(B): Wireless Sensor Networks

Course Objectives:

- ☐ To know the basics of Wireless Sensor Networks (WSN).
- ☐ To familiarize with the Architecture of WSN.
- ☐ To understand the concepts of Networking in WSN.
- ☐ To understand the applications of WSN

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Understand challenges and technologies for WSN.
2. Understand Sensors.
3. Understand the MAC, Routing, and Transport Control Protocols for WSN
4. Understand the infrastructure establishment for WSN.

Unit-I:

Overview of Wireless Sensor Networks: Introduction, Architectural elements, Basic Technology, Sensor type, Operating systems and Middleware, Transport protocols, Data dissemination and routing types.

Sensor Network organization and tracking, Computation, Communication, Sensor Node hardware and software, Sensor node technology, Sensor taxonomy, Characteristics and design challenges. Applications of WSN.

Unit-II:

Fundamentals of MAC protocols: Performance requirements, Common Protocols. MAC Protocols for WSN: Schedule based protocols, Random access-based protocols, Low Duty Cycle Protocols and Wake Up Concepts, Sensor-MAC Case Study, B-MAC, IEEE 802.15.4 LR-WPANs Standard Case Study.

Unit-III:

Routing Protocols for WSN: Data dissemination and gathering, Challenges and Design issues, Routing strategies: techniques, flooding & variants, protocols for information via negotiation, Low-energy Adaptive clustering hierarchy, Power-efficient gathering, Directed diffusion, Geographical routing.

Unit-IV:

Transport Control Protocols for WSN: Design Issues, Existing Protocols: CODA, (Congestion Detection and Avoidance), ESRT (Event-to-Sink Reliable Transport), RMST (Reliable Multi-Segment Transport), PSFQ (Pump slowly and fetch quickly), GARUDA, ATP (Ad Hoc Transport Protocol). Performance: Congestion and packet loss recovery.

Text Books:

- ☐ K. Sohraby Minoli and T. Zanti, “Wireless Sensor Networks: Technology, Protocols, and Applications”, John Wiley and Sons, March 2007.
- ☐ H. Karl, and A. Willig, “Protocols and Architectures for Wireless Sensor Networks”, John Wiley and Sons, October 2007.

Reference Book:

Feng Zhao & Leonidas J. Guibas, “Wireless Sensor Networks- An Information Processing Approach”, Elsevier, 2007.

Major-1-P - XIX(C) : Cyber Security

Minor-1-P-IV (Without Research)

Course Objectives:

- To understand the growing importance of cyber security.
- To understand the use of crypto systems.
- To learn about various security issues and cyber crimes

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Have basic understanding of security concerns
2. Learn about cryptography
3. Learn about various software security issues
4. Know about different cyber crimes

Unit-I:

Introduction: Computer Security Concepts, Threats, Attacks, and Assets, Security Functional Requirements, Fundamental Security Design Principles. Confidentiality, Integrity, Availability, Computer Criminals, Motive of attackers, active attacks, passive attacks, Software attacks, hardware attacks, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage, etc., Comprehensive Cyber Security Policy.

Unit-II:

Cryptography: History and development of Cryptography. Substitution and affine ciphers, Transpositions Cipher, Confusion, diffusion, Symmetric, Asymmetric Encryption. DES, Modes of DES, Uses of Encryption, Hash function, key exchange, Digital Signatures, Digital Certificates.

Unit-III:

Software Security: Types of Malicious Software, Advanced Persistent Threat, Propagation —Infected Content - Viruses, Propagation — Vulnerability Exploit - Worms, Propagation — Social Engineering — SPAM E-Mail, Trojans, Payload — System Corruption, Payload — Attack Agent — Zombie, Bots, Payload — Information Theft — Keyloggers, Phishing, Spyware, Payload — Stealthing — Backdoors, Rootkits, Countermeasures.

Network Security: Denial-of-Service Attacks, Flooding Attacks, Distributed Denial-of- Service Attacks, Overview of Intrusion Detection, Honeypots, The Need for Firewalls

Unit-IV:

Classification of cyber crimes, Common cyber crimes- cyber crime targeting computers and mobiles, cyber crime against women and children, financial frauds, social engineering attacks, malware and ransomware attacks, zero day and zero click attacks, Cybercriminals modus-operandi , Reporting of cyber crimes, Remedial and mitigation measures, Legal perspective of cyber crime, IT Act 2000 and its amendments, Cyber crime and offences, Organisations dealing with Cyber crime and Cyber security in India, Case studies.

Text Books:

- ✓ W. Stallings, L. Brown, *Computer Security: Principles and Practice*, 4th edition, Pearson Education, 2018.
- ✓ Nina Godbole and Sunit Belpure, *Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives*, Wiley
- ✓ C. P. Pfleeger, S. L. Pfleeger; *Security in Computing*, Prentice Hall of India, 2006
- ✓ Marc H. Rogers, *"Cybersecurity: A Practical Guide to the Law of Cyber Risk"*, Wiley

SEMESTER- VIII

Major-1-P -XX: Cloud Computing (Without Research)

Major-1-P -XIX: Cloud Computing (Without Research)

Course Objectives:

- ☐ To familiarize with basic concepts of cloud as a computing paradigm
- ☐ To learn about different cloud services

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn about the evolution cloud computing and the concept of virtualization
2. Understand the cloud migration process and design issues
3. Familiarize with different service models
4. Understand different security issues and disaster management services

Unit-I:

Overview of Computing Paradigm - Recent trends in Computing - Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing - Evolution of cloud computing - Cloud Computing (NIST Model) Characteristics - Pros and Cons of Cloud Computing, Cloud computing vs. Cluster computing vs. Grid computing – Role of Open Standards - Cloud computing stack - Service Models (XaaS).

Unit-II:

Cloud Computing Architecture: Layers in cloud architecture, Software as a Service (SaaS), features of SaaS and benefits, Platform as a Service (PaaS), features of PaaS and benefits, Infrastructure as a Service (IaaS), features of IaaS and benefits, Service providers, challenges and risks in cloud adoption. Cloud deployment model: Public clouds – Private clouds – Community clouds - Hybrid clouds.

Unit-III:

Infrastructure as a Service (IaaS) – Introduction- IaaS definition, virtualization, Different approaches to virtualization, Hypervisors, Machine Image, Virtual Machine (VM) - Resource Virtualization – Server, Storage, Network, Virtual Machine(resource) provisioning and manageability, Data storage in cloud computing (storage as a service) - Examples - Amazon EC2 - Renting, EC2 Compute Unit.

Platform as a Service (PaaS) – Introduction, Service Oriented Architecture (SOA) - Cloud Platform and Management – Computation, Storage – Examples - Google App Engine, Microsoft Azure, Salesforce.com, Force.com platform - Software as a Service (SaaS) - Introduction to SaaS - Web services, Case Study on SaaS.

Unit-IV:

SLA Management in Cloud Computing: Traditional Approaches to SLO Management, Types of SLA, Life Cycle of SLA, SLA Management in Cloud, Pricing Mechanism, SLA Violation.

Cloud Security - Infrastructure Security - Network level security, Host level security, Application- level security – Data security and Storage - Data privacy and security Issues, Jurisdictional issues raised by Data location - Identity & Access Management - Access Control - Trust, Reputation, Risk - Authentication in cloud computing, Client access in cloud, Cloud contracting Model.

Text Books:

- ☐ Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, *Cloud Computing: Principles and Paradigms*, Wiley.
- ☐ *Cloud Computing*, U S Pandey & K Choudhary, S. Chand Pubs.
- ☐ *Cloud computing a practical approach* - Anthony T.Velte , Toby J. Velte Robert Elsenpeter TATA McGraw- Hill , New Delhi.

Reference Books:

- ☐ Thomas Erl, Ricardo Puttini, Zaigham Mahmood, *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall.
- ☐ Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, *Mastering Cloud Computing*, Mc Graw Hill Education.
- ☐ *Cloud Computing: Principles, Systems and Applications*, Editors: Nikos Antonopoulos, Lee Gillam, Springer.

Major-1-P -XX Lab: Cloud Computing (Without Research)

Major-1-P -XIX Lab: Cloud Computing (With Research)

1. Study of computing paradigms and comparison of Grid, Cluster, Distributed, Utility and Cloud Computing
2. Analysis of cloud computing architecture and NIST cloud model characteristics
3. Study of cloud service models (IaaS, PaaS, SaaS) and XaaS framework

4. Comparison of cloud deployment models (Public, Private, Community, Hybrid)
5. Virtualization in cloud computing: implementation and study of hypervisors and virtual machines
6. Resource provisioning in IaaS using virtualization tools (VM creation, allocation and management)
7. Hands-on with Amazon EC2: launching and managing virtual instances
8. Study and implementation of PaaS platforms (Google App Engine / Microsoft Azure basics)
9. Case study and implementation of SaaS applications and web services
10. Cloud security and SLA management: study of authentication, access control, and SLA violation analysis

Tools/Platforms Suggested

- AWS (Amazon Web Services EC2)

Major-1-P -XXI: Machine Learning (Without Research)

Major-1-P -XX: Machine Learning (With Research)

Course Objectives:

- To understand the concept of machine learning for intelligent data processing
- To learn various classification and clustering techniques to analyze data

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn Decision tree learning algorithms.
2. Learn neural network and hypothesis accuracy estimation.
3. Apply Supervised Learning to obtain a predicted output.
4. Apply Unsupervised Learning on data.

Unit-I:

Introduction: Brief Introduction to Machine Learning Supervised Learning Unsupervised Learning Reinforcement Learning, Overview of supervised learning, classification, and regression problems, Statistical Pattern Recognition: Bayes Decision Theory, Minimum Error and Minimum Risk Classifiers, Discriminant Function and Decision Boundary, Normal Density, Discriminant Function for Discrete Features. Naïve Bayes Classification.

Unit-II:

Classification: K-nearest neighbourhood (KNN) classifier, variation of k-NN classifiers. Decision tree learning, Issues in Decision tree learning.

Artificial Neural Network: Introduction – Fundamental concept – Evolution of Neural Networks – Basic Models of Artificial Neural Networks – Important Terminologies of ANNs
– McCulloch-Pitts Neuron – Linear Separability – Back-Propagation Network – Radial Basis Function Network, multi-level classification.

Unit-III:

Model Assessment and Selection: Loss function, test and training error, Bias, Variance, and model complexity, Bias-variance trade off, Bayesian approach and BIC, Cross-validation, Boot strap methods, Performance of Classification algorithms (Confusion Matrix, Precision, Recall and ROC Curve), Introduction to Gradient Descent, The Curse of Dimensionality, Dimensionality Reduction, Principal Component Analysis.

Unit-IV:

Unsupervised Learning and Random forests: Cluster analysis (k-means, Hierarchical clustering, DBSCAN, spectral clustering), Gaussian mixtures and EM algorithm, Random forests and analysis.

Introduction to Deep Learning, Case studies on Digit classification and Image recognition.

Text Books:

- ✓ Tom M. Mitchell, *Machine Learning*, McGraw Hill Education, Indian Edition.
- ✓ Alpaydin, E., *Introduction to Machine Learning*. United Kingdom: MIT Press.
- ✓ S.N. Sivanandan and S.N. Deepa, *Principles of Soft Computing*, Wiley India.

Reference Books:

- ✓ C. M. Bishop – *Pattern Recognition and Machine Learning*, Springer.
- ✓ Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning-Data Mining, Inference, and Prediction*, Second Edition, Springer Verlag.

Major-1-P -XXI-Lab: Machine Learning (Without Research)

Major-1-P -XX-Lab: Machine Learning (With Research)

1. Write Python programs for Creation and Loading different types of datasets in Python using the required libraries.
 - a) Creation using pandas
 - b) Loading CSV dataset files using Pandas
 - c) Loading datasets using sklearn
2. Write a python program to compute Mean, Median, Mode, Variance, Standard Deviation using Datasets
3. Demonstrate various data pre-processing techniques for a given dataset. Write a python program to compute
 - a) Reshaping the data,
 - b) Filtering the data,
 - c) Merging the data
 - d) Handling the missing values in datasets
 - e) Feature Normalization: Min-max normalization
4. Implement Naïve Bayes theorem to classify the English text using Python.
5. Write a python program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6. Implement linear regression using python.
7. Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Vary the number of k values as follows and compare the results:
 - a. 1
 - b. 3
 - c. 5
 - d. 7
 - e. 11
8. Build Artificial Neural Network model with back propagation on a given dataset.
9. Implement Dimensionality reduction using Principal Component Analysis (PCA) method on a dataset (For example Iris).
10. Write a python program to implement K-Means clustering Algorithm.
11. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering.
12. Write a python program to implement DBSCAN clustering Algorithm.
13. Consider different classification problem like Digit Classification, Image Classification. Implement that using different ML techniques and compare their result and make a report.

Major-1-P -XXII(A): Compiler Design (Without Research)

Major-1-P -XXI(A): Compiler Design (With Research)

Course Objectives:

- To learn the process of translating a modern high-level language to executable code.
- To understand the fundamental principles in compiler design
- To apply optimization techniques to have better code generation

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Learn the process of compiling and lexical analysis
2. Understand syntax analysis and use of different parsers
3. Understand Syntax Directed Translation and code generation
4. Learn about the issues in the design of a code generator

Unit-I:

Introduction to Compiling: Compilers, Analysis of the source program, The phases of a compiler, Cousins of the compiler, The grouping of phases, Compiler-construction tools.

Lexical Analysis: The role of the lexical analyzer, Input buffering, Specification of tokens, Recognition of tokens, A language for specifying lexical analyzers, Finite automata, From a regular expression to an NFA, Design of a lexical analyzer generator.

Unit-II:

Syntax Analysis: The role of the parser, Review of Context-Free Grammars – Derivation trees and Parse Trees,

Ambiguity, eliminating ambiguity, Left recursion, Left factoring. Top- Down Parsing: Recursive Descent parsing, Predictive parsing, LL(1) Grammars. Bottom-Up Parsing: Shift Reduce parsing, LR parsing – Constructing SLR parsing tables, Constructing, Canonical LR parsing tables and Constructing LALR parsing tables.

Unit-III:

Syntax Directed Translation: Syntax Directed Definitions, Evaluation orders for SDD's, Application of SDT, SDT schemes, Implementing L-attribute SDD's.

Intermediated Code Generation: Need for intermediate code, Types of intermediate code, Three address code, Quadruples, Triples, Assignment statements, Boolean expressions.

Unit-IV:

Run-Time Environments: Source Language issues, Storage organization, Storage allocation strategies, Access to nonlocal names, parameter passing, Symbol tables.

Code generation: Issues in the design of a code generator, Basic blocks and flow graphs, A Simple code generator, Register allocation and assignment, The DAG representation of basic blocks, Peephole optimization, Generating code from DAGs.

Text Books:

- ✓ *Compilers Principles, Techniques and Tools* by Alfred V. Aho, Monica S. Lam, RaviSethi, Jeffrey D. Ullman, Pearson.
- ✓ *Compiler Design* by Santanu Chattopadhyay, PHI.

Reference Book:

Engineering a Compiler by Keith D. Cooper & Linda Torczon, Morgan Kaufmann.

Major-1-P -XXII(B): : Internet of Things (Without Research)

Major-1-P -XXI(B): : Internet of Things (WithResearch)

Course Objectives:

- To know the evolution of the Internet of Things (IoT), its working mechanisms.
- To know and develop Applications of IoT in our daily life.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Understand IoT networking components and various topologies
2. Understand IoT connectivity & communication technologies
3. Understand functions of Arduino, Raspberry Pi and other platforms
4. Understand various application domains and develop small sensor-based applications

Unit-I:

Emergence of IoT: Introduction, Evolution of IoT, Layered Architecture, Networking Components, Addressing Strategies. IoT Enabling Technologies: Wireless Sensor Networks, Cloud Computing, Big Data Analytics, Embedded Systems.

IoT Levels and Deployment Templates: Introduction, IoT Level-1 to Level-6. IoT Sensing and Actuation: Introduction, Sensors, their characteristics and types. Actuators, their characteristics and types.

Unit-II:

Application Domains of IoT: Introduction, Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health & Lifestyle. Paradigms, Challenges and Future: Evolution of new IoT Paradigms, Challenges associated with IoT, Emerging Pillars of IoT.

IoT Processing Topologies: Data Format, importance of processing, processing topologies, IoT device design and selection considerations, Processing offloading.

Unit-III:

IoT Connectivity Technologies: Introduction & types, IEEE 802.15.4, Zigbee, Thread, Wireless HART, RFID, NFC, Z-Wave, LoRa, Wi-Fi, Bluetooth.

IoT Communication Technologies: Introduction, Protocols for Infrastructure, Discovery, Data, Identification, Device Management and Semantic.

Unit-IV:

Introduction to various IoT Development Platforms, Sensor development boards/platforms, Arduino versus Raspberry Pi. Arduino: Introduction, installation & setup. Introduction to Sketch, Data Types, Operators, Control Statements, Arrays, String, Common Functions and Libraries.

Text Books:

- ✓ *Introduction to IoT* by S Mishra, A. Mukharjee & A. Roy, Cambridge University Press.

- ✓ *Internet of Things: A hands-on approach* by A. Bahga & V. Madiseti, University Press.

Reference Books:

- ✓ *The Internet of Things: Enabling Technologies, Platforms, and Use Cases* by Pethuru Raj and Anupama C. Raman, CRC Press
- ✓ *D. Hanes, G. Salgueiro, P. Grossetete, R. Barton, J. Henry; IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things*, PHI.

Major-1-P -XXII(C) : Introduction to Data Science (Without Research)

Major-1-P -XXI(C) : Introduction to Data Science (With Research)

Minor-1-P-V(Without Research)

Course Objectives:

- To understand emerging issues related to various fields of data science.
- To understand the underlying principles of data science, exploring data analysis.
- To learn the basics of R Programming.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Appreciate the importance of data science & learn the use different data analysis tools
2. Learn R Programming
3. Understand the techniques for data cleaning
4. Learn the use of various data analysis and visualization tools

Unit-I:

Data Scientist's Tool Box: Turning data into actionable knowledge, introduction to the tools that are used in building data analysis software: version control, markdown, git, GitHub, R, and RStudio.

Unit-II:

R Programming Basics: Overview of R, R data types and objects, reading and writing data, Control structures, functions, scope rules, dates and times, Loop functions, debugging tools, Simulation, code profiling.

Unit-III:

Getting and Cleaning Data: Obtaining data from the web, from APIs, from databases and other sources in various formats, basics of data cleaning and making data "tidy".

Unit-IV:

Exploratory Data Analysis: Essential exploratory techniques for summarizing data, applied before formal modeling commences, eliminating or sharpening potential hypotheses about the world that can be addressed by the data, common multivariate statistical techniques used to visualize high-dimensional data.

Text Book:

Rachel Schutt, Cathy O'Neil, "Doing Data Science: Straight Talk from the Frontline" Schrott/O'Reilly, 2013.

Reference Books:

- ✓ Foster Provost, Tom Fawcett, "Data Science for Business" What You Need to Know About Data Mining and Data-Analytic Thinking by O'Reilly, 2013.
- ✓ John W. Foreman, "Data Smart: Using data Science to Transform Information into Insight" by John Wiley & Sons, 2013.
- ✓ Eric Seigel, "Predictive Analytics: The Power to Predict who Will Click, Buy, Lie, or Die", 1st Edition, by Wiley, 2013.

Major-1-P -XXII (C) Lab: Introduction to Data Science (Without Research)

Major-1-P -XXI (C) Lab: Introduction to Data Science (With Research)

Minor-1-P-V Lab: Introduction to Data Science (Without Research)

1. Study of basic Syntaxes in R
2. Implementation of vector data objects operations
3. Implementation of matrix, array and factors and perform variance analog in R
4. Implementation and use of data frames in R
5. Create Sample (Dummy) Data in R and perform data manipulation with R
6. Study and implementation of various control structures in R
7. Data Manipulation with dplyr package
8. Data Manipulation with data.table package

9. Study and implementation of Data Visualization with ggplot2
10. Study and implementation data transpose operations in R

Major-1-P-XXIII: Project Work-II (Without Research)

Student of four-year Bachelor's Degree (Honours without Research) are required to take up a software development project under the guidance of a faculty Member. The students are expected to initiate the project work during seventh semester and complete the Project in the Eighth Semester and submit a project report for evaluation by an external examiner. The template for project report can be as **Major-1-P - XV: Project Work-I**.

The evaluation pattern of the project will be as follows:

- i. Problem formulation and definition
- ii. Execution of code & results
- iii. Documentation
- iv. Clarity in presentation
- v. Performance in the Viva voce

For 4-Year Hons. with Research

Research Methodology & Dissertation(12 Credits) (With Research)

(Research Methodology: 4 credits + Dissertation: 8 credits)

Student of four-year Bachelor's Degree (Honours with Research) are required to take up Research Projects under the guidance of a faculty Member. The students are expected to initiate the project work during seventh semester and complete the Research Project in the Eighth Semester. The Research outcome of their project work may be published in peer-reviewed journals or may be presented in conferences/ seminars or may be patented. The evaluation of the dissertation has to be made by an external examiner.

Major-1-P-XXII : Research Methodology (With Research)

Course Objectives:

- To understand some basic concepts of research and its methodologies To identify appropriate research topics
- To select and define appropriate research problem To prepare a project proposal
- To write a research report and thesis

Learning Outcomes:

Upon completion of this course, students will be able to:

1. Demonstrate the ability to choose methods appropriate to research aims and objectives
2. Understand the limitations of particular research methods
3. Develop skills in qualitative and quantitative data analysis
4. Learn about various data analysis techniques using various programming languages and different software tools

Unit-I:

Research Methodology: An Introduction: Meaning and objective of Research Methodology, Motivation in research, types of research, Research Approaches, significance of research, Research Methods versus Methodology, Research and Scientific Method, Importance of Knowing How Research is Done, Research Process, Criteria of Good Research, Problems Encountered by Researchers in India.

Defining the Research Problem: Research Problem, Selecting the Problem, Necessity of Defining the Problem, Technique Involved in Defining a Problem, Illustration and Conclusion

Unit-II:

Research Design: Meaning of Research Design, Need for Research Design, Features of a Good Design, Important Concepts Relating to Research Design, Different Research Designs, Basic Principles of Experimental Designs and Conclusion.

Methods of Data Collection: Collection of Primary Data, Observation Method, Interview Method, Collection of Data through Questionnaires, Collection of Data through Schedules, Difference between Questionnaires and Schedules, Some Other Methods of Data Collection, Collection of Secondary Data, Selection of Appropriate Method for Data Collection.

Unit-III:

Processing and Analysis of Data: Processing Operations, Some Problems in Processing, Elements/Types of Analysis, Statistics in Research, Measures of Central Tendency, Measures of Dispersion, Measures of Asymmetry, Measures of

Relationship, Simple Regression Analysis, Multiple Correlation and Regression, Curve Fitting.

Testing of Hypotheses: What is a Hypothesis, Basic Concepts Concerning Testing of Hypotheses, Procedure for Hypothesis Testing, Important Parametric Tests, Hypothesis Testing of Means, Hypothesis Testing for Comparing Two Related Samples, Hypothesis Testing of Proportions, Hypothesis Testing for Difference between Proportions.

Unit-IV:

Chi-square Test: Chi-square as a Test for Comparing Variance, Chi-square as a Non-parametric Test, Conditions for the Application of χ^2 Test, Steps Involved in Applying Chi-square Test, Alternative Formula, Yates' Correction, Conversion of χ^2 into Phi Coefficient, Conversion of χ^2 into Coefficient by Contingency, Important Characteristics of χ^2 Test.

Analysis of Variance: Analysis of Variance (ANOVA), The Basic Principle of ANOVA, ANOVA Technique

Books:

- ✓ *Research Methodology Methods and Techniques*, Kothari, C. R., New Age.
- ✓ *Research Methodology: a step-by-step guide for beginners*, Kumar, Pearson Education.
- ✓ *Practical Research Methods*, Dawson, C., UBSPD Pvt. Ltd.
- 5. *Research Methodology*, Sharma, N. K., KSK Publishers, New Delhi

MULTIDISCIPLINARY COURSES

A. Computer Fundamental

Course Objectives:

- ☐ Introduce number systems and data representation
- ☐ Understand functional units and components of computer
- ☐ Introduce the emerging technologies

Learning Outcomes:

Upon completion of this course, students will be able to:

- ☐ Understand the basic organization of a computer and the number system
- ☐ Learn about the working of commonly used input-output and memory devices
- ☐ Understand the role of Operating system and Computer Networks
- ☐ Know about some of the emerging computing technologies and web services

UNIT-1:

Computer Basics: Simple Model of a Computer, Characteristics of Computers, Hardware and Software, working of a Computer, Stored Program Concept, Problem Solving with computer: Flowchart, Algorithms, Programming,

Computer Software: Introduction to computer software, classification of computer software, system software, application software, firmware, middleware

UNIT-2:

Input/output Units: Input devices, Output devices, Computer Memory: Introduction, Read Only Memory, Serial Access Memory, Cache memory, primary memory, secondary storage devices, magnetic tapes, hard disks, SSD, optical drives, USB flash drivers, Memory cards, Mass storage devices, Memory Hierarchy.

UNIT-3:

Operating Systems: Definition, Batch Operating System, Multiprogramming Operating System, Time Sharing Operating System, Multiprocessing Operating System. Services of OS.

Computer Networks: Concepts of Networking-LAN, WAN, MAN, Network topologies.

Internet and the World Wide Web.

UNIT-4:

Emerging Computing Environments: Peer to Peer Computing, Grid computing, distributed computing, Cloud Computing: Introduction, cloud services, cloud deployment models.

Email, video conferencing, e-Learning, e-Banking, UPI, e-commerce, e-Governance, social networking, emerging computer applications.

Text Book:

- ☐ *Fundamentals of Computers* by V Rajaraman 6th edition PHI Learning Private Limited

Reference Books:

- ☐ *A First Course in Computers* by Sanjay Saxena, Vikas Publishing House.
- ☐ *Computer Fundamentals* by Anita Goel, Pearson pub

B. Introduction to Web Technologies

Course Objectives:

- To learn the fundamentals of web designing.
- To design and develop standard and interactive web pages.

Learning Outcomes:

Upon completion of this course, students will be able to:

- Understand Internet, Internet Protocols, and World Wide Web
- Understand HTML and its tags
- Learn the design and development of web pages
- Learn the styles/layouts of web pages using CSS & client-side scripting using JavaScript

UNIT-1:

- Introduction to Internet, Internet Protocols, World Wide Web (WWW): Introduction, History, HTTP, Web Browser, Web Server with example, Web page, working principles of WWW. Web Development: Introduction, Front-end and Backend Development Technologies. Concepts of Client-Server communication.
- Introduction to HTML: Introduction, Characteristics, Advantages and Disadvantages of HTML, HTML Editors, Understanding elements in HTML, Container and empty elements, Basic Tags and Attributes: <!DOCTYPE>, <HTML>, <HEAD>, <TITLE>, <BODY>, <P>, Attributes of the basic tags. Creating a Simple HTML Web Page, running a web page in the browser.

UNIT-2:

- Working with HTML Tags: Headings, Break, Horizontal Line. Formatting Text with HTML Elements: Italic, Bold, Small, Subscript, Superscript and changing background color. Adding Comments in HTML. Working with Hyperlink, List, Table, Image, Video, and Frames in HTML.
- Creating Forms in HTML: Form Tag and its attributes. Creating Text box, Password box, Text area, Drop-down list, Checkbox, Radio button, Email, Color picker, Date-Time picker, Submit and Reset buttons.

UNIT-3:

- Cascading Style Sheets (CSS): Introduction, Benefits of using CSS, Understanding the Syntax, CSS Selectors, Using CSS: External, Internal Inline CSS. Comments in CSS,
- Basic CSS Properties: Color, Background, Text, Font, List, Display. CSS Box Model: Introduction, working with Margin, Border, and Padding. Working with CSS Navigation Bar and Drop-Downs.

UNIT-4:

- JavaScript: Introduction, Features, Benefits, Creating Simple JavaScript. Using JavaScript in HTML: Use in the Head, Body and as external script file. Exploring Popup Boxes: alert, confirm, prompt box. Display Possibilities: innerHTML, document.write(), window.alert(), console.log().
- Programming using JavaScript: Introduction to Data types, Variables, Operators, Expressions (Arithmetic, String, Logical), Comments. Control Statements: Selection Statements: if, if...else, nested if...else, else...if ladder, switch. Loops: while, do...while, for. Jump Statements: break, continue. Functions in JavaScript: built-in and user defined, Invoking a function, scope of a function, global vs local variables.

Text Book:

- *Web Technologies (Black Book)*, DreamTech Press

Reference Books:

- *Web Enabled Commercial Application Development Using HTML, JavaScript, DHTML and PHP 4th Edition* by Ivan Bayross.
- *HTML, XHTML and CSS Bible, 5ed*, Wiley India-Steven M. Schafer.

C. Internet and Ethical Practices

Course Objectives:

- To learn about different Internet services
- To understand the security and ethical issues in the use of Internet

Learning Outcomes:

Upon completion of this course, students will be able to:

- Understand Internet and World Wide Web
- Understand about the security issues and use of cryptographic methods
- Know about the security challenges in e-commerce
- Know about the growing security concerns in the cyberspace and legal perspectives to ensure ethical use of Internet

UNIT-1:

Data Communication over Internet, connecting to the Internet, broadband, leased connection, WWW, Web browsers, URL, HTTP, Blogs, Search engines, e-mail services, teleconferencing, social networks, online services,

UNIT-2:

Network Security and Cryptography: Cryptography, Cryptography terminologies, CIA, Security goals, Attacks, Security services, Types of cryptography-symmetric and asymmetric key cryptography (basics only), Cryptanalysis, Steganography.

UNIT-3:

Security issues in E-Commerce: Types of E-Commerce models (B2B, B2C, C2B, C2C), EFT (Electronic Fund Transfer), Modes of EFT, Online Payment and The IT Act 2000, IPR in Cyberspace.

UNIT-4:

Security Challenges in Cyberspace: Hacking, Child Pornography, Cyber Stalking, Denial of service, Malware, Phishing, Classification of cyber-crimes, Common cyber-crimes, cyber-crime targeting computers and mobiles, cyber-crime against women and children, financial frauds, social engineering attacks, Legal perspective of cyber-crime, Organizations dealing with Cybercrime and Cyber security in India.

Text Books:

- *Computer Fundamentals and Applications*, Ashok Arora, Vikas Pubs.
- *Cryptography and Network Security*, Behrouz A. Forouzan, Debdeep Mukhopadhyay

Reference Books:

- Henry Chan, Raymod Lee and et al., “*E-Commerce Fundamental and Applications*”, Wiley
- C. P. Pfleeger, S. L. Pfleeger, *Security in Computing*, Prentice Hall of India.

SKILL ENHANCEMENT COURSES (SEC)

A. Advanced Python Programming

Course Objective:

This course introduces students to data visualization in the field of exploratory data science using Python.

Learning Outcomes:

Upon completion of this course, students will be able to:

- Learn use of objects & classes and perform data handling using Numpy arrays
- Do data manipulation using Pandas

UNIT-1:

- Object Oriented Programming: classes, objects and methods.
- File and exception handling: File handling through libraries; Errors and exception handling.
- Array Manipulation using Numpy: Numpy array: Creating Numpy arrays, Data Types for Numpy arrays, Arithmetic with NumPy Arrays Basic Indexing and Slicing.

UNIT-2:

Data Manipulation using Pandas: Data Structures in Pandas: Series, DataFrame, Index objects, Loading data into Pandas data frame. Working with Data frames. Grouped and aggregate calculations.

Text Books:

- McKinney W. *Python for Data Analysis: Data Wrangling with Pandas, NumPy and IPython*. 2nd edition. O’ Reilly Media, 2018.

- *Chen D. Y, Pandas for Everyone: Python Data Analysis, Pearson, 2018.*
- *Balaguruswamy E. Introduction to Computing and Problem Solving using Python, 2nd edition, McGraw Hill Education, 2018*

Lab: Advanced Python Programming

1. Write a Python class named Person with attributes name, age, weight (kgs), height (ft) and takes them through the constructor and exposes a method get_bmi_result() which returns one of "underweight", "healthy", "obese".
2. Write a python program to demonstrate various kinds of inheritance.
3. Write a python program to catch following exception i) Value Error ii) Index Error iii) Name Error iv) Type Error v) Divide Zero Error
- 4.a) Create a numpy array from list, tuple with float type
- b) Python program to demonstrate slicing, integer and boolean array indexing
5. a) Write a python program to find min, max, sum, cumulative sum of array.
- b) Write a python program to demonstrate use of ndim, shape, size, dtype.
6. a) Write a python program to implement Pandas Series with labels.
- b) Create a Pandas Series from a dictionary.
- c) Creating a Pandas DataFrame.
- d) Write a program which makes use of following Pandas methods
i) describe () ii) head() iii) tail()
7. a) Write a program that converts Pandas DataFrame and Series into numpy.array.
- b) Write a program that demonstrates the column selection, column addition, and column deletion.
- c) Write a program that demonstrates the row selection, row addition, and row deletion.
- d) Get n-largest and n-smallest values from a particular column in Pandas dataframe
8. a) Write a program which use pandas inbuilt visualization to plot following graphs:
i. Histograms ii. Line plots iii. Scatter plots iv. Bar plots
- b) Write a program to demonstrate use of groupby() method.
9. a) Write a program to demonstrate pandas Merging, Joining and Concatenating
- b) Creating dataframes from csv and excel files.
10. Write a Python program using pandas that finds Missing Data and replace missing data.
5. Write a Python class named Person with attributes name, age, weight (kgs), height (ft) and takes them through the constructor and exposes a method get_bmi_result() which returns one of "underweight", "healthy", "obese".
6. Write a python program to demonstrate various kinds of inheritance.
7. Write a python program to catch following exception i) Value Error ii) Index Error iii) Name Error iv) Type Error v) Divide Zero Error
8. a) Create a numpy array from list, tuple with float type
- b) Python program to demonstrate slicing, integer and boolean array indexing
5. a) Write a python program to find min, max, sum, cumulative sum of array.
- b) Write a python program to demonstrate use of ndim, shape, size, dtype.
6. a) Write a python program to implement Pandas Series with labels.
- b) Create a Pandas Series from a dictionary.
- c) Creating a Pandas Data Frame.
- d) Write a program which make use of following Pandas methods
i) describe() ii) head() iii) tail()
7. a) Write a program that converts Pandas Data Frame and Series into numpy.array.
- b) Write a program that demonstrates the column selection, column addition, and column deletion.
- c) Write a program that demonstrates the row selection, row addition, and row deletion.
- d) Get n-largest and n-smallest values from a particular column in Pandas data Frame
- 8.a) Write a program which use pandas' inbuilt visualization to plot following graphs:
i. Histograms ii. Line plots iii. Scatter plots iv. Bar plots
- b) Write a program to demonstrate use of group by() method.
9. a) Write a program to demonstrate pandas Merging, Joining and Concatenating
- b) Creating data frames from csv and excel files.
10. Write a Python program using pandas that finds Missing Data and replace missing data.

B. Principles of Management

Course Objectives:

- To understand the basic principles of management
- To provide an insight into different management functions and strategies

Learning Outcomes:

Upon completion of this course, students will be able to:

- Understand the evolution management and various school of thoughts
- Learn different management functions and decision-making process
- Know about different leadership styles and importance of co-ordination
- Learn about the need for strategic management

Unit I:

Nature of Management: Meaning, Definition, its nature purpose, importance & Functions, Management as Art, Science & Profession- Management as social System Concepts of management-Administration- Organization. Evolution of Management Thought: Contribution of F.W. Taylor, Henri Fayol, Elton Mayo, Chester Barhard & Peter Drucker to the management thought. Various approaches to management (i.e. Schools of management thought) Indian Management Thought.

Unit II:

Functions of Management (Part-I) Planning - Meaning - Need & Importance, types levels– advantages & limitations, Forecasting - Need & Techniques, Decision making - Types - Process of rational decision making & techniques of decision making. Organizing - Elements of organizing & processes: Types of organizations, Delegation of authority - Need, difficulties in delegation – Decentralization.

Unit III:

Functions of Management (Part-II) Staffing - Meaning & Importance, Direction - Nature – Principles, Communication - Types & Importance, Motivation - Importance – theories, Leadership - Meaning - styles, qualities & functions of leaders. Controlling- Need, Nature, importance, Process & Techniques, Coordination - Need, Importance.

Unit IV:

Strategic Management Definition, Classes of Decisions, Levels of Decision, Strategy, Role of different Strategist, Relevance of Strategic Management and its Benefits, Strategic Management in India.

Text Books:

- *Essential of Management* by Horold Koontz & Itenz Weibrich, McGraw-Hills International.
- *Essential of Business Administration* by K. Aswathapa, Himalaya Publishing House.

Reference Books:

- *Principles & Practice of Management* by L.M. Prasad, Sultan Chand & Sons pub.
- *Principles of Management* by Tripathi & Reddy, Tata McGraw Hill

VALUE AIDED COURSES (VAC)

A. Environmental Studies & Disaster Management

Unit 1: Multidisciplinary nature of environmental studies (12 Period)

Definition, scope and importance

Need for public awareness

Environmental Pollution

Definition

• Cause, effects and control measures of:-

- a) Air pollution
- b) Water pollution
- c) Soil pollution
- d) Marine pollution
- e) Noise pollution
- f) Radiation pollution

Unit 2: Natural Resources: (12 Period)

Renewable and non-renewable resources:

Natural resources and associated problems.

- a) Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people.
- b) Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems.
- c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies.
- d) Food resources: World food problems, changes caused by agriculture and Overgrazing, effects of modern agriculture, fertilizer-pesticide problems, waterlogging, salinity, case studies.

- e) Energy resources : Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. Case studies.

Biodiversity:-

Introduction-Definition;Biogeographically classification of India

India as a mega diversity nation. Hot spots of biodiversity, Threats to biodiversity. Endangered and endemic species of India. Conservation of biodiversity. In Situ and Ex-situ conservation of biodiversity

Unit-3: Disaster Management (12 Period)

- 1) **Disaster Management:** Types of disasters (natural and Man-made) and their causes and effect)
- 2) **Vulnerability Assessment and Risk analysis:** Vulnerability to various disasters(Flood, Cyclone, Earthquake, Heat waves, Desertification and Lighting)
- 3) **Institutional Framework:** Institutional arrangements for disaster management (National Disaster Management Authority (NDMA), State Disaster Management Authority (SDMA), Disaster Management Act, 2005, District Disaster Management Authority(DDMA), National Disaster Response Force(NDRF) and Odisha Disaster Rapid Action Force(ODRAF)
- 4) **Preparedness measures:** Disaster Management cycle, Early Warning System, Pre-Disaster and Post-Disaster Preparedness, strengthening of SDMA and DDMA, Community Preparedness for flood cyclone, heat waves, fire safety, lightening and snake biting. Stakeholders participation, Corporate Social Responsibility (CSR)
- 5) **Survival Skills:** Survival skills adopted during and after disaster (Flood, Fire, Earthquake, Cyclone and Lightening), Disaster Management Act-2005, Compensation and Insurance

Unit 4: Social Issues and the Environment (9Period)

A.

- a) Environmental Ethics: Issues and possible solutions.
 - b) Climate change, global warming, acid rain, ozone layer depletion, nuclear
 - c) accidents and holocaust. Case studies
 - d) Environment Protection Act
 - e) Air(Preservation Control of Pollution) Act
 - f) Water(Preservation Control of Pollution) Act
 - g) Wildlife Protection Act
 - h) Forest Conservation Act
- Solid waste management Cause, effect and Control Measure of Urban and Industrial waste
(Role of each individual in conservation of Natural resources and prevention of pollution)

B. Human Population and the Environment

Population Ecology: Individuals, species, population, community
Human population growth, population control method
Urbanisation and its effect on society

Unit 5: Field work (15 Periods of 30 hrs)

- Visit to an area to document environmental assets: river/forest/flora/fauna, etc.
- Visit to a local polluted site- Urban/Rural/Industrial/Agricultural
- Study of common plants, insects, birds and basic principles of identification.

B. Digital Fluency

Course Objectives

- ☐ To develop skills for locating, evaluating, and using digital information effectively and responsibly.
- ☐ To introduce common digital tools and technologies including word processing, presentation software, and spreadsheets.
- ☐ To emphasize responsible use of social media and online communities and the implications of digital footprints
- ☐ To explore effective communication techniques in digital environments, such as email etiquette and online messaging

Teaching Learning Process

- ☐ Lecture
- ☐ Assignment
- ☐ Individual and Group Presentation

Learning Outcomes

- ☐ To understand the importance of evaluating the credibility of online sources and proper citation and referencing of digital

content

- Enhanced skills in using digital tools to create, learn, and share content online effectively
- Have imbibed values important in digital work including privacy, security, and intellectual property rights
- Able to use collaborative tools and platforms for managing online discussions, virtual meetings, and group projects

Unit-I: Digital Literacy

Locating, evaluating and using digital information online. Evaluating the credibility and reliability of online sources.

Proper

citation and referencing of digital sources

Unit-II: Digital Capabilities

Introduction to common digital tools and technology. Use of digital tools to learn, create, and share online. Word processing,

presentation software, spreadsheets.

Unit-III: Digital Principles

Values when working digitally. Digital privacy and security. Intellectual property rights and copyright laws in the digital world. Responsible use of social media and online communities.

Unit-IV: Digital Communication and Collaboration

Effective communication in digital environments (email etiquette, online messaging, etc.) Collaborative tools and platforms for

teamwork and group project. Managing online discussions and virtual meetings

C. Intellectual Property Rights (IPR)

The value-added course in Intellectual Property Rights (IPR) is designed to provide

Undergraduate students with a comprehensive understanding of the legal frameworks and Principles governing intellectual property. Through an exploration of patents, copyrights, trademarks, and international conventions, students will gain insights into the protection and management of intellectual creations across various domains.

COURSE OBJECTIVES:

- To familiarize students with the fundamental concepts of intellectual property, including patents, copyrights, trademarks, and their characteristics.
- To explore the legal frameworks and international agreements governing intellectual property rights, such as the Berne Convention, Paris Convention, Madrid Agreement, and TRIPS.
- To equip students with knowledge of the essentials of patents, including registration procedures, rights of patentees, and issues related to infringement and compulsory licensing.
- To enable students to understand copyright law, covering topics such as the subject matter of copyright, ownership rights, economic and moral rights, fair use, and infringement.

COURSE OUTCOMES:

- Students will demonstrate a clear understanding of the basic concepts and nature of intellectual property rights, including patents, copyrights, and trademarks.
- Students will be able to analyze and interpret the provisions of international conventions and agreements related to intellectual property rights and their implications for legal protection.
- Students will develop practical skills in navigating the registration procedures and understanding the rights and obligations associated with patents, copyrights, and trademarks.
- Students will gain the ability to evaluate legal issues and apply relevant legal principles to real-world scenarios involving intellectual property disputes, infringement claims and protection strategies.

LEARNING OUTCOMES:

- Understand the rationale and importance of intellectual property protection in fostering innovation, creativity, and economic development.
- Analyze the legal frameworks and international agreements governing intellectual property rights and their impact on global trade and commerce.
- Apply legal principles to assess the eligibility, registration, and enforcement of patents, copyrights, and trademarks in various industries and sectors.
- Develop critical thinking skills to evaluate the ethical, social, and economic implications of intellectual property rights on society, culture, and technology.

COURSE STRUCTURE

Unit I Introduction to IPR

- (a) Basic concept of Intellectual Property
- (b) Characteristics and Nature of Intellectual Property right
- (c) Justifications for protection of IP
- (d) International Conventions and agreements - Berne Convention, Paris Convention, Madrid Agreement, TRIPS

Unit II Patents

- (a) Essentials of patents
- (b) Registration and term of patent
- (c) Rights of patentee, Compulsory licenses and Government use of patent
- (d) Infringement

Unit III Copyright

- (a) Subject matter of Copyright
- (b) Ownership of copyright
- (c) Rights of owner-Economic Rights, Moral Rights
- (d) Fair Use and Infringement

Unit IV Trademark

- (a) Trademark: Conventional and Non-conventional Marks, Concept of distinctiveness
- (b) Absolute and relative grounds of refusal
- (c) Procedure for registration and Term of protection
- (d) Infringement and Passing Off

SUGGESTED BOOKS:

- ☐ *Intellectual Property: Patents, Copyright, Trademarks and allied rights* - W R Cornish, Universal Law Publishing
- ☐ *Indian Patent Law and Practice* - Kalyan C. Kankanala, Arun K. Narasani & Vinita Radhakrishnan; Oxford University Press
- ☐ *Patent law essentials: a concise guide* - Durham, Alan L - Quorum Books
- ☐ *Law relating to patents, trademarks, copyright designs, geographical indications* - BL Wadehra, Universal Law Publishing
- ☐ *Intellectual property; patents, trademarks and copyrights* - Stim, Richard Thomson
- ☐ *Learning*
- ☐ *Patent law* – P Narayanan Eastern Law House
- ☐ *Intellectual Property Law*, Lionel Bently & Brad Sherman, Oxford.
- ☐ *Intellectual Property Law*, P. Narayanan, Eastern Law House

D. Social Statistics and Computer Applications

Statistics is numerically expressed. In those numbers the social scientist tries to find trends of events so that shape critical thinking, evaluate policies, and take informed decisions. Numbers do not speak for themselves, but they need to be contextualized to unravel degrees of inequality and vulnerability among groups. They can be used for better understanding every aspect of society by finding trends among human population.

Course Outcome:

This course would help students develop skills on social statistical techniques and computer applications enhancing their chances of employability.

Unit I: Measures of Central Tendency

- ☐ Data: Meaning, Nature, Types
- ☐ Mean: Computation, Types, Merits and Limitations
- ☐ Median: Computation, Partition Value, Merits and Limitations
- ☐ Mode: Computation, Properties, Merits and Limitations

Learning Outcomes:

- ☐ Measures of central tendency informs us on a value that works as a reference point for comparing and understanding human population better. Students would learn to calculate the measures of central tendency.

Unit II Measures of Dispersion

- ☐ Mean Deviation
- ☐ Standard Deviation
- ☐ Coefficient of Variation
- ☐ Lorenz Curve and Ginni's Coefficient of concentration

Learning Outcomes:

Measures of dispersion informs us on the limitations of measures of central tendency and tells us on the significance of working on variations around a central value.

Unit III Skewness, Kurtosis Correlation and Regression Analysis

- Skewness, Kurtosis
- Bivariate distributions, Characteristics of Association,
- Chi square, t-test, Analysis of Variance,
- Correlation and Regression.

Learning Outcomes:

- Correlation and regression are statistical measurements that informs us about the definite relation between two variables. This unit would help students develop the skills of calculating them and understanding the nature of relationship between variables.

Unit IV Computer application in social science research

- Introduction to Computer Applications: MS Word, Excel, Powerpoint.
- Macro data analysis and computer application: Excel, Stata.
- Methods and use of macro-statistics and secondary sources: Analysis of Data published by Census of India, NSSO.
- Application of Computers in Social research (e.g. SPSS).

Learning Outcomes:

- This unit helps students learn about the basic computer applications that macro data and data from secondary sources could be analysed. This unit can provide the skills to social science students to use computer applications for social science research.

Text Books:

- Babbie, E. R. (2020). *The practice of social research*. United States: Cengage.
- Judith Handel. 1978. *Introductory Statistics for Sociology*. NJ: Prentice- Hall Inc.
- Babbie, E., Halley, F., & Zaino, J. (2007). *Adventures in social research: data analysis using SPSS 14.0 and 15.0 for Windows*. Pine Forge Press.
- Singh, A.K. (2022). *Tests, measurements and research methods in behavioural sciences*. New Delhi: Bharati Bhawan Publishers and Distributors.

Reference Books:

- Anthony Capon, J.1988. *Elementary Statistics for the Social Sciences*. Wadsworth Publishers.
- Bryman, Alan. 1988. *Quality and Quantity in Social Research*, London: Unwin.
- De Vaus, David. 2002. *Analysing Social Science Data: 50 Key Problems in Data Analysis*, New Delhi, Sage Publications.
- Irvine, J., I. Miles and J. Evans (eds.) 1979. *Demystifying Social Statistics*, London: Pluto Press.
- Madge, John. 1970. *The Origins of Scientific Sociology*. London: Tavistock.
- Marsh, Catherine. 1988. *Exploring Data*. Cambridge: Polity Press

E-resources:

1. Computer Applications in Social Science Research, IGNOU.

Link: <https://egyankosh.ac.in/bitstream/123456789/63507/2/Unit-16.pdf>

VOCATIONAL COURSES

A. Office Automation Tools

Course Objectives:

The course introduces the students to document processing, presentation software and data handling. The basic features and skills of creating, editing, inserting tables, graphics as well as presentation tools along with spreadsheet data handling are covered.

Learning Outcomes:

Upon completion of this course, students will be able to:

- Create and refine documents using text formatting, tables and graphics.
- Use mail merge.
- Create macros and templates in documents.
- Create presentations with transitions, animations and other advanced presentation features

UNIT-1:

Document Processing Basics: Creating, opening and saving a document, text formatting, header and footer, creating and editing of tables, importing graphics, insert picture, using word processor's drawing features, text in drawing.

UNIT-2:

Advanced Features: Creating macros, watermarks, templates, reviewing documents, comparing and combining documents, protection of documents-using passwords. Mail merge concept, main document, data sources, merging data source and main document.

UNIT-3:

Presentation Tools: Creating presentations, using blank presentation option, using design template option, adding slides, deleting a slide, importing images from the outside world, deleting a slide, numbering a slide, saving presentation transition and animations, adding notes to slides, customize slideshows.

UNIT-4:

Data Handling using spreadsheets: Working with spreadsheets, formatting the sheets, Relative, Absolute and mixed referencing, mathematical and statistical functions, nested functions. **Charts:** Data visualization using built-in charts.

Text Book:

□ *LibreOffice 6.0 Writer Guide. Friends of OpenDocument, Inc., 2018.*

Lab: Office Automation Tools

1. Create a professional resume using a word processor. Include the following elements in your resume: Your personal information (name, contact details, Photo), A section for your education background, A section for your work experience, Proper text formatting (e.g., font type, size, bold, italics), Utilize headers and footers to include your name and page numbers
2. Create a report for a company's quarterly sales performance. Use a word processor to: Insert a table to organize the sales data (e.g., sales figures by region), Format the table with appropriate borders and shading, Include a header with the report title, Insert a footer with the current date and page numbers, Import a company logo or relevant graphics to enhance the report's visual appeal.
3. As part of a project presentation, you need to create a slide deck using a presentation software. Your presentation should include: Text slides summarizing key project milestones and objectives, Use of bullet points to highlight important information, Incorporate relevant graphics or images to complement your content, Utilize the software's drawing features to annotate diagrams or charts, Ensure consistent formatting and design throughout the presentation.
4. Prepare a newsletter for your organization using a word processor. Your newsletter should feature: Multiple columns for easier readability, Insertion of images or clipart to accompany articles, Incorporate text boxes for quotes or side notes, Include hyperlinks to external websites or email addresses for more information, Utilize header and footer sections for consistency and branding.
5. Create a standardized template for project proposals in your organization. Develop a template using a word processor that includes: Customized headers and footers with organization logo and document title, Watermark feature to indicate the document status (e.g., draft, final), Protection of the document using a password to prevent unauthorized editing, Reviewing feature enabled to track changes and comments from multiple reviewers, Utilize macros to automate repetitive tasks (e.g., inserting common sections or formatting).
6. Create a presentation for an upcoming sales of a company. Use presentation software to: Choose a suitable design template from the software's library, Add slides to cover key topics such as company overview, product/service offerings, and client testimonials, Import images or graphics from external sources to enhance the visual appeal of the presentation, Customize slide numbering to maintain a coherent structure throughout the presentation, Save the presentation with transition effects and animations to make it engaging for the audience.
7. Prepare a training presentation for new employees on company policies and procedures. Use presentation software to: Start with a blank presentation and organize slides to cover various policy topics such as code of conduct, safety protocols, and HR guidelines, Utilize a design template that reflects the professional and informative nature of the training material, Import relevant images or diagrams to illustrate key points and make the presentation visually engaging, Customize slide transitions to ensure smooth progression between topics and maintain audience interest, Add speaker notes to each slide to provide trainers with additional information and guidance during the presentation.
8. Organize sales data for the past quarter into a spreadsheet and create a summary report for management. Use spreadsheet software to: Create a new spreadsheet and format the sheet to include relevant column headers for sales data categories such as date, product name, quantity sold, and total revenue, Enter the sales data for each day of the quarter and use relative referencing to calculate the total revenue for each product, Apply appropriate formatting to the cells to enhance readability and make important data stand out, Use mathematical functions such as SUM and AVERAGE to calculate the total revenue and average sales quantity for the quarter, Create a nested function to identify the best-selling product based on total revenue, and present the summary report using built-in charts for visual representation of sales trends.
9. Analyze sales data for a marketing campaign and identify the most effective advertising channels for driving sales. Use spreadsheet software to: Import sales data into a new spreadsheet and format the sheet to include columns for advertising channel, sales revenue, and conversion rate, Use relative referencing to calculate the conversion rate for each advertising channel based on the number of sales and leads generated, Apply conditional formatting to highlight advertising channels with high conversion rates or low sales revenue for further analysis, Utilize nested functions to calculate the total sales revenue and average conversion rate across all advertising channels, Generate charts such as line graphs or scatter plots to visualize the relationship between advertising spend, conversion rate, and sales revenue, helping you identify the most cost-effective advertising channels for the marketing campaign.

B. Multimedia and Animation

Course Objectives:

To learn the use of various tools and techniques to create multimedia and animation. This will give an opportunity to the students to use their creativity for developing interesting multimedia and animated objects which can be used for advertising and gaming industry.

Learning Outcomes:

Upon completion of this course, students will be able to:

- ☐ Understand the tools and techniques used for multimedia creations
- ☐ Create different graphics
- ☐ Add audio visual effects
- ☐ Learn the use of animation tools

UNIT-1:

Introduction: Definition Multimedia and uses, applications, Hardware and Software requirements, multimedia presentation and authoring tools, LAN and multimedia, multimedia servers and database: vector graphics, 3-D graphics program, animation techniques, shading, anti-aliasing, morphing: video on demand.

UNIT-2:

Text and Graphics: Uses of text in multimedia, families and faces of fonts, outline fonts, bitmap fonts, International characters sets and hypertext digital font techniques. Graphics- Introduction, types, color and color models, color palettes, vector drawing, 3-D drawing and rendering, making still images, editing and capturing images

UNIT-3:

Audio and Video: Digital representation of sound, MIDI audio, MIDI vs. Digital Audio, Audio file formats, Adding sound to your multimedia project, Audio software and hardware, Video basic, how video works, broadcast video standard, Analog video, Digital video, shooting and editing video, video compression and file formats.

UNIT-4:

Animation and Multimedia Authoring: Introduction, principles, types and uses, cell animation, computer animations, morphing, Animation software compression techniques: need and types of compression- lossy and lossless, CODECs, authoring basic, types of authoring tools. Multimedia on the Web: Introduction, Bandwidth, text on the web- Dynamic and embedded font technology, Audio and video on the web, buffering and streaming, webcasting, videoconferencing.

Text Books:

- ☐ *Multimedia: Making it work by Tay Vughan, TataMcGraw-Hill*
- ☐ *Computer Graphics and Multimedia by A. P. Godse, Technical Publication*
- ☐ *Multimedia: An Introduction, Villamil and Molina, PHI*
- ☐ *Sound and Video, Lozano. Multimedia, PHI*

Lab: Multimedia and Animation

1. Using Inkscape, create a vector graphic logo for a fictional company, incorporating text and geometric shapes.
2. Develop a multimedia presentation using LibreOffice Impress, including text slides, images, audio clips, and simple animations.
3. Edit a digital photograph using GIMP, applying basic adjustments such as cropping, color correction, and retouching.
4. Record and edit a short audio clip using Audacity, adding effects like echo or noise reduction to enhance the sound.
5. Shoot a short video using a smartphone or digital camera, then edit it using Shotcut or OpenShot video editor.
6. Create a motion graphics animation using Synfig Studio, incorporating vector graphics, text, and simple effects.
7. Design a 3D character model using MakeHuman, then animate it using Blender's rigging and animation tools.
8. Create an interactive multimedia quiz using H5P, incorporating text, images, audio, and video elements with interactive questions.
9. Enhance the promotional image to make it more visually appealing using GIMP. Use the Levels tool to adjust the brightness and contrast, apply the Sharpen filter to enhance the sharpness and clarity of important details, use Crop tool to remove any unnecessary elements or empty space from the edges of the image, experiment with the Color Balance tool to adjust the color tones, Add text overlays using the Text tool to convey essential information about the event, such as the date, time, and location. Incorporate visual effects and filters, such as blurs or gradients, to create depth and visual interest in specific areas of the image. Use the Export function to save the edited image in a web-friendly format (e.g., JPEG or PNG) suitable for online promotion and distribution.
10. Design a simple 3D model using Blender, then render it to create a still image or animation sequence. Create a multimedia presentation for a marketing campaign. Use the open-source software Blender to: Incorporate vector graphics and 3-D graphics to enhance the visual appeal of the presentation. Apply animation techniques such as morphing and shading to create dynamic transitions between slides. Utilize antialiasing to smooth out jagged edges and improve the overall image quality. Export the presentation as a video file compatible with video-on-demand platforms for distribution.
11. Produce an audio podcast for an online radio show. Use the open-source software Audacity to: Record and edit digital audio tracks, including voiceovers, music, and sound effects for the podcast. Convert MIDI audio files to digital audio format and synchronize them with other audio elements in the podcast. Export the final podcast episode in popular audio file formats suitable for streaming and downloading on the web.

12. Develop an animated short film for a film festival. Use the open-source software Synfig Studio to: Create cell animations and computer-generated animations to bring characters and scenes to life in the short film. Experiment with morphing techniques to seamlessly transition between different character poses and facial expressions. Apply compression techniques to optimize the file size of the animated short film without sacrificing visual quality. Export the final animated short film in standard video file formats for submission to the film festival.

C. Introduction to Web Design

Course Objectives:

- ☐ To learn the fundamentals of web designing.
- ☐ To learn front-end web designing.
- ☐ To design and develop standard and interactive web pages.

Course Outcomes:

Upon completion of this course, students will be able to:

- ☐ Understand Internet, Internet Protocols, and World Wide Web
- ☐ Understand HTML and its tags
- ☐ Learn the design and development of web pages
- ☐ Learn the styles and layouts of web pages using CSS

UNIT-1:

- ☐ Introduction to the Internet, Internet Protocols, World Wide Web (WWW): Introduction, History, HTTP and HTTP methods, Web Browser, Web Server and their examples, Web page, working principles of WWW. Web Development: Introduction, Front-end and Back-end Development Technologies. Concepts of Client-Server communication.
- ☐ Introduction to HTML: Introduction, History, Characteristics, Advantages and Disadvantages of HTML, HTML Editors, Understanding elements in HTML, Container and empty elements, Basic Tags and Attributes: `<!DOCTYPE>`, `<HTML>`, `<HEAD>`, `<TITLE>`, `<BODY>`, `<P>`, Attributes of the basic tags. Creating a Simple HTML Web Page, running a web page in the browser.

UNIT-2:

- ☐ Working with HTML Tags: Headings, Break, Horizontal Line. Formatting Text with HTML Elements: Italic, Bold, Small, Subscript, Superscript and changing background color. Adding Comments in HTML. Working with Hyperlink, List, Table, Image, Video, and Frames in HTML.
- ☐ Creating Forms in HTML: Form Tag and its attributes. Creating Text box, Password box, Text area, Drop-down list, Checkbox, Radio button, Email, Color picker, Date- Time picker, Submit and Reset buttons.

UNIT-3:

Cascading Style Sheets (CSS): Introduction, Benefits of using CSS, Understanding the Syntax, CSS Selectors, Using CSS: External, Internal Inline CSS. Comments in CSS, Basic CSS Properties: Color, Background, Text, Font, List, Display. CSS Box Model: Introduction, working with Margin, Border, and Padding. Working with CSS Navigation Bar and Drop-Downs.

UNIT-4:

- ☐ JavaScript: Introduction, Features, Benefits, Creating Simple JavaScript. Using JavaScript in HTML: Using in the Head, Body and as external script file. Exploring Popup Boxes: alert, confirm, prompt box. Display Possibilities: inner HTML, document.write(), window.alert(), console.log().
- ☐ Programming using JavaScript: Introduction to Data types, Variables, Operators, Expressions (Arithmetic, String, Logical), Comments. Control Statements: Selection Statements, Loops, Jump Statements. Functions in JavaScript: built-in and user defined, Invoking a function, scope of a function, global vs local variables. Introduction to DOM, Event handling and Form validation in JavaScript.

Lab: Introduction to Web Design

1. Create a Web Page to display “Hello HTML”.
 - ☐ Display the same using different headings: h1 to h6
 - ☐ Apply bold, italic formats
 - ☐ Change text color, background of the page
2. Create a Web Page to display the list of Fruits using both ordered and unordered lists. Recreate the lists using images of the same items.
3. Use the web page created in Question Number 2. When the user clicks on the image of a Fruit, it should open a new page that contains an image of the Fruit along with its benefits (use multiple paragraphs, and make sure the image is aligned properly).
4. Create a web page that displays details of the Fruits in Tabular format. Use serial no., name, color, taste, price/kg.
5. Create a Web Page that displays a video file (record/create a video of your own).
 - ☐ Apply various controls such as play, pause, volume.
 - ☐ Apply autoplay, muted and both.
 - ☐ Display a Youtube Video in your Web Page.

6. Design a Student registration form to collect various data about a Student which includes Name, Age, Gender (M, F, O), Mobile No., Email ID, Stream (Science/Arts/Commerce in drop-down), Choice for participating in NCC, NSS, YRC (use check box), and two buttons for reset and submit respectively. Display the form at the center of the page with proper alignment of each item in the form.
7. Use the web page created in Question no. 4. Use CSS and apply various styling to the text, colors to each row of the table, styling to borders and background color of the table.
8. Create a Web page with a Horizontal Navigation bar containing four items such as Home, College, Students, Teachers. The first item should be active, by default. The background color of the item changes when the user moves the mouse over it.
9. Create a Web Page that asks the user to enter the number of Students, and then iteratively ask the details of each Student and display them in a list, using JavaScript.
10. Modify the above program (Q. 9) to display the details in a tabular format, dynamically using JavaScript.
11. Create a Web Page with two text fields and four buttons to perform arithmetic operations such as Addition, Subtraction, Multiplication and Division. The user has to enter numbers in the text fields and press any of the above buttons. The JavaScript program should perform corresponding operations and display the result in the same page. [NOTE: Use different functions for each operation]
12. Use JavaScript and validate the form data in the Student registration Page (created in Q. no. 6). When the user clicks the submit button, the program has to validate that every field contains valid data. [NOTE: validate name, age, gender, email, mobile number]
13. Write a JavaScript program to ask the user to enter a number
 - a. Find factorial
 - b. Find sum of numbers from one to that number
 - c. Find number of prime numbers present from one to that number
14. Write a JavaScript program to demonstrate the use of mouse and keyboard events.

D. Data Analysis using Spread sheet

Course Objectives:

The objective is to enhance the knowledge of statistics and enable students to make sense of data by analyzing and visualizing it using spreadsheets, interpreting the results and gaining insights.

Learning Outcomes:

Upon completion of this course, students will be able to:

- ☐ Learn the use different functions in spreadsheets
- ☐ Do data analysis using basic and advanced statistical functions
- ☐ Learn the use of tools for visualization
- ☐ Do complex level data analysis

UNIT-1:

Data Handling: Spreadsheet concepts, Managing worksheets, Formatting cells, Entering data, Handling operators in formula, Cell referencing and naming of cells and cell ranges, Sorting, Multilayer sorting, Data validation, Find and Replace, Paste special, Filter and advanced filter, Formatting as table, Pivot tables, Formulae vs functions, Cell formulae vs Array formulae. Mathematical functions, Statistical functions, Logical functions, Date and Time functions, Lookup and reference: Hlookup, and Vlookup, Index and Match functions, Text functions. What-if-analysis: Goal-seek, Data tables, Scenario manager.

UNIT-2:

Data Analysis: Explore a data model: its content, and its structure, using the Power Pivot addin. Learning DAX formula language. Create calculated fields and calculated measure for each cell, filter context for calculation, and explore several advanced DAX functions. Cube formulas to retrieve data from data model.

UNIT-3:

Data Visualization: Different types of charts including Pivot charts: Column, Line, Pie, Bar, Scatter charts. Fine tuning of charts: Chart Elements, Chart Styles, Chart Filters, Box Plot.

UNIT-4:

Utilities and Protection: What-if scenarios, goal-seek, solver, data validation, creating a dropdown list from a range of cells, data filtering and sorting, calculations using linked sheets, detective tools. Passwords and digital signatures in Spreadsheets

Books:

- ☐ Mali, L., Libre Office 5.1 Writer, Calc: Math Formula Book - Vol 1. Notion Press
- ☐ Libre Office 7.0 Calc Guide.
- ☐ Goldmeier, J., Advanced Excel Essentials, Apress, 2014
- ☐ Schmuller, J., Statistical Analysis with Excel for Dummies, 4th Edition, Wiley India

Lab: Data Analysis using Spread sheet

1. You are given a dataset containing sales data for a retail company. Perform the following tasks using spreadsheet software: Format the cells in the "Sales Amount" column as currency with two decimal places. Use the SUM function to calculate the total sales amount for the dataset.
2. You are managing inventory data for a retail store. Use mathematical functions to calculate the total value of the inventory based on the unit price and quantity of each item in stock.
3. Analyze the sales performance data for a company over the past year. Use statistical functions to calculate the mean, median, and standard deviation of monthly sales figures to identify trends and fluctuations.
4. Create a spreadsheet to track student attendance. Use logical functions to automatically mark students as "Present" or "Absent" based on their attendance record, considering specific criteria such as late arrivals or early departures.
5. Develop a project timeline for a software development project. Use date and time functions to calculate project milestones, deadlines, and durations based on the start date and estimated completion times for each task.
6. Analyze student performance data. Use the provided dataset and perform the following actions: Create a pivot table to summarize the average scores for each subject across different classes. Apply data validation to restrict entries in the "Grade" column to only A, B, C, D, or F.
7. Utilize DAX formula language to calculate the total revenue generated by each product category. Apply filter context to analyze revenue trends over different time periods.
8. Create a pivot chart to visualize the sales performance of different product categories over the last quarter. Customize the chart elements and styles to enhance clarity and visual appeal.
9. Use the provided dataset to create a column, line, pie, box plot chart to analyze the distribution of student scores across multiple subjects.
10. Implement what-if scenarios to analyze the impact of changes in interest rates on loan repayments. Use the solver tool to find the optimal solution for minimizing loan payments.
11. Set up data validation to create a dropdown list of employee names from a range of cells. Apply password protection to restrict access to sensitive financial data stored in the spread sheet.