

# DEPARTMENT OF COMPUTER SCIENCE

## SYLLABUS OF UG PROGRAMME (B.Sc.)



SKILLS	
EMPLOYABILITY	
ENTERPRENURESHIP	

**RAMA DEVI WOMEN'S UNIVERSITY**  
Vidya Vihar, Bhubaneswar-751022, Odisha  
Website: <https://rdwu.ac.in>

**DEPARTMENT OF COMPUTER SCIENCE  
SYLLABUS OF UG PROGRAMME  
(B.Sc)**



**RAMA DEVI WOMEN'S UNIVERSITY  
VIDYA VIHAR, BHUBANESWAR  
ODISHA**

*Ms. Fanki*  
21.10.23  
Controller of Examinations  
R.D. Women's University  
Bhubaneswar

**DEPARTMENT OF COMPUTER SCIENCE**  
**Rama Devi Women's University, Bhubaneswar**

**B.Sc. Computer Science**

**Programme Outcomes (POs)**

- PO1. Scientific knowledge:** Apply the knowledge of science, mathematics, and computing to the solution of complex scientific problems.
- PO2. Problem analysis:** Identify, formulate, research literature, and analyze complex scientific problems reaching substantiated conclusions using principles of mathematics, natural sciences, and applied sciences.
- PO3. Design/development of solutions:** Design solutions for complex problems and design system processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- PO4. Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- PO5. Modern tools usage:** Create, select, and apply appropriate techniques, resources, and modern computing and IT tools including prediction and modeling to complex scientific activities with an understanding of the limitations.
- PO6. The software engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional practice.
- PO7. Environment and sustainability:** Understand the impact of the professional software engineering solutions in societal and environmental contexts, and demonstrate the need for sustainable development.
- PO8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the scientific practice.
- PO9. Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- PO10. Communication:** Communicate effectively on complex activities with the scientific community and with the society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- PO11. Project management:** Demonstrate the scientific and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- PO12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broad context of technological change.

## Programme Specific Outcomes

A graduate with a B.Sc. in Computer Science will have the ability to

**PSO1.** Demonstrate in the following core knowledge areas

- Data Structures and Programming Languages
- Databases, Software Engineering and Development
- Computer Architecture and Security

**PSO2.** Apply problem-solving skills and the knowledge of computer science to solve real world problems.

**PSO3.** Develop technical project reports and present them orally among the users

## COMMON SYLLABUS FOR BSc COMPUTER SCIENCE

### B. Sc. (Honours) Computer Science (CBCS)

SEME STER	COURSE OPTED	COURSE NAME	CREDITS
<b>I</b>	Ability Enhancement Course-1	AECC-1 (Environmental Science)	2
	Core Course-1	Programming using C	4
	Core Course-1 Practical	Programming using C LAB	2
	Core Course-2	Digital Logic	4
	Core Course-2 Practical	Digital Logic Lab	2
	Generic Elective-1	GE-1	4
	Generic Elective-1 Practical	GE-1 Tutorial/ LAB	2
<b>II</b>	Ability Enhancement Course-2	AECC-2 (English Communication/MIL)	2
	Core Course-3	Programming using C++	4
	Core Course-3 Practical	Programming using C++ LAB	2
	Core Course-4	Data Structures	4
	Core Course-4 Practical	Data Structures LAB	2
	Generic Elective-2	GE-2	4
	Generic Elective-2 Practical	GE-2 Tutorial/ LAB	2
<b>III</b>	Core Course-5	JAVA Programming	4
	Core Course-5 Practical	JAVA Programming LAB	2
	Core Course-6	Database Systems	4
	Core Course-6 Practical	Database Systems LAB	2
	Core Course-7	Discrete Mathematical Structures	4
	Core Course-7 Practical	Discrete Mathematical Structures LAB	2
	Skill Enhancement Course-1	SEC-1	2
	Generic Elective-3	GE-3	4

	General Elective-3 Practical	GE-3 Tutorial/ LAB	2
<b>IV</b>	Core Course-8	Operating Systems	4
	Core Course-8 Practical	Operating Systems LAB	2
	Core Course-9	Computer Networks	4
	Core Course-9 Practical	Computer Networks LAB	2
	Core Course-10	Computer Graphics	4
	Core Course-10 Practical	Computer Graphics LAB	2
	Skill Enhancement Course-2	SEC-2	2
	Generic Elective-4	GE-4	4
	General Elective-4 Practical	GE-4 Tutorial/ LAB	2
<b>V</b>	Core Course-11	Web Technology	4
	Core Course-11 Practical	Web Technology LAB	2
	Core Course-12	Software Engineering	4
	Core Course-12 Practical	Software Engineering Lab	2
	Discipline Specific Elective-1	DSE-1	4
	Discipline Specific Elective-1 Practical	DSE-1 LAB/ Tutorial	2
	Discipline Specific Elective-2	DSE-2	4
	Discipline Specific Elective-2 Practical	DSE-2 LAB/ Tutorial	2
<b>VI</b>	Core Course-13	Artificial Intelligence	4
	Core Course-13 Practical	Artificial Intelligence LAB	2
	Core Course-14	Algorithm Design Techniques	4
	Core Course-14 Practical	Algorithm Design Techniques LAB	2
	Discipline Specific Elective-3	DSE-3	4
	Discipline Specific Elective-3 Practical	DSE-3 LAB/ Tutorial	2
	Discipline Specific Elective-4	DSE-4	4
	Discipline Specific Elective-4 Practical	DSE-4 LAB/ Tutorial	2

**CORE Papers:(Credit: 06 each)**

**CORE–1: Programming Using C**

**CORE – 2: Digital Logic**

**CORE – 3: Programming Using C++**

**CORE – 4: Data Structure**

**CORE – 5: Operating Systems**

**CORE – 6: Database Systems**

**CORE – 7: Discrete Mathematical Structures**

**CORE – 8: Java Programming**

**CORE – 9: Computer Network**

**CORE – 10: Computer Graphics**

**CORE – 11: Web Technologies**

**CORE – 12: Software Engineering**

**CORE – 13: Artificial Intelligence**  
**CORE – 14: Algorithm Design Techniques**

**Discipline Specific Electives (DSE) Papers: (Credit: 06 each)**

**DSE–1: Numerical Techniques**  
**DSE–2: Unix Programming**  
**DSE–3: Elementary Data Science**  
**DSE–4: Project Work/ Dissertation OR Data Mining**

**Skill Enhancement Courses (SEC): (Credit: 02 each)**

**SEC – 1: Android Programming.**  
**SEC – 2: Programming in Python.**

**Ability Enhancement Courses (AEC): (Credit: 02 each)**

**AEC– 1: Environmental Science.**  
**AEC – 2: English Communication/MIL.**

**Generic Elective (GE): (Credit: 06 each)** papers offered by Computer Science/IT Departments for other disciplines. It is recommended that the other departments must offer the following papers as GE.

**GE – 1: Computer Fundamentals**  
**GE – 2: C and Data Structures**  
**GE – 3: Programming in Python**  
**GE – 4: Web Technology**

However the students from **Computer Science/IT** disciplines shall choose **four papers of any one discipline** as their GE papers from the following list.

**GE-1:**

- a) Mathematics–1
- b) Physics–1
- c) Statistics–1
- d) Electronics –1

**GE-2:**

- a) Mathematics–2
- b) Physics–2
- c) Statistics–2
- d) Electronics –2

**GE-3:**

- a) Mathematics–3
- b) Physics–3
- c) Statistics–3
- d) Electronics –3

**GE-4:**

- a) Mathematics–4
- b) Physics–4
- c) Statistics–4
- d) Electronics –4

**Detailed Syllabus****CORE – 1: Programming Using C**

---

**Course Outcomes:**

After completing this course, students will be able to:

**CO1.** Understand fundamentals of C, learn various programming constructs, and write C programs using operators and control structures.

**CO2.** Develop C programs with pointers and arrays, perform pointer arithmetic.

**CO3.** Apply code reusability with functions, demonstrate dynamic memory allocation, and command line arguments.

**CO4.** Handle files using several file handling mechanisms, solve problems using derived data types.

**Unit-1**

**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) **Control structures:** Decision Making and Branching (Simple IF Statement, IF...ELSE Statement, Nesting IF... ELSE Statement, ELSE IF Ladder), Selection control structure (Switch Statement).

**Unit-2**

**Loops:** The WHILE Statement, The DO...WHILE Statement, The FOR Statement, Jumps in Loops, **Array:** Concept of Array, Array Declaration, types of array (one and multiple dimension), Character Arrays and Strings, Subscript and pointer representation of array, Array of Pointers, Limitation of array, **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.

**Unit-3**

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

**Function:** Function & types (User defined function, library function) Function Definition, Declaration, Function Calls, Header file and library, Function Arguments, string handling function (strlen, strcmp, strcpy, strncpy, strcat, strstr), Function recursion, Functions Returning Pointers, Pointers to Functions, Command line arguments, Application of pointer (dynamic memory allocation).

#### **Unit-4**

**Structure and Union:** Defining, Declaring, Accessing, Initialization Structure, nested structure, self-referential structure, bit-field, Arrays of Structures, Structures and Functions, Unions, difference between structure and union, active data member, structure within union, Self-referential Structure, **File:** File Management in C, Defining and Opening a File, File opening modes (read, write, append), Closing a File, File operations, file and stream, Error Handling During I/O Operations, sequential and random access file, low level and high level file.

#### **Text Books:**

1. E. Balagurusamy, "Programming in ANSI C", 4/e, (TMH)

#### **Reference Books:**

1. B. Kernighan & Dennis Ritchie, "The C Programming Language", 2/e PHI
2. Paul Deitel, Harvey Deitel, "C: How to Program", 8/e, Prentice Hall.

#### **Core-1 Practical: Programming Fundamentals using C Lab**

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 no. 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 using macro.
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 orderThe 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 alter 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.

#### Mapping of Course Outcomes with the Program Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	4	2	2	2	1	2	1	2	1	2	2
CO2	5	4	4	4	2	1	2	1	2	1	2	2
CO3	5	4	5	4	2	1	2	1	4	1	4	2
CO4	5	4	4	4	2	1	2	1	4		2	2

## CORE-2: DIGITAL LOGIC

---

### Course Outcomes

**CO1:** Able to define different logic gates, illustrate realization of Boolean expression in SOP and POS form and design it using logic gates

**CO 2:** Able to design the logic circuits like adder, subtractor etc.

**CO 3:** Become accustomed to design and test combinational circuits

**CO 4:** Become accustomed to design and develop sequential circuits

### **Unit-1**

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-2**

Arithmetic: Addition and Subtraction of Signed Numbers, Addition/ Subtraction Logic Unit, Design of Fast Adders: Carry-Lookahead Addition, Multiplication of Positive Numbers, Signed-Operand Multiplication: Booth Algorithm, Fast Multiplication: Bit-Pair Recoding Multipliers, Carry-Save Addition of Summands, Integer Division, Floating-Point Numbers and Operations: IEEE Standard for Floating-Point Numbers, Arithmetic Operations on Floating-Point Numbers, Guard Bits and Truncation, Implementing Floating-Point Operations.

### **Unit-3**

Flip-Flops, Gated Latches, Master-Slave Flip-Flops, Edge-Triggering, T Flip-Flops, JK Flip-Flops. Registers and Shift Registers, Counters, Decoders, Multiplexers, Programmable Logic Devices (PLDs), Programmable Array Logic (PAL), Complex Programmable Logic Devices (CPLDs), Field-Programmable Gate Array (FPGA), Sequential Circuits, UP/ DOWN Counters, Timing Diagrams, The Finite State Machine Model, Synthesis of Finite State Machines.

### **Unit-4**

Memory System: Semiconductor RAM Memories, Internal Organization of Memory Chips, Static Memories, Asynchronous DRAMS, Synchronous DRAMS, Structure of Large Memories, Memory System Considerations, RAMBUS Memory. Read-Only Memories: ROM, PROM, EPROM, EEPROM, Flash Memory, Speed, Size, and Cost of Memory. Secondary Storage: Magnetic Hard Disks, Optical Disks, Magnetic Tape Systems.

### **Text Books:**

1. Carl Hamacher, Z. Vranesic, S. Zaky: Computer Organization, 5/e (TMH)

### **Reference Books:**

1. M. Morris Mano: Digital Logic and Computer Design, Pearson

**CORE-2 Practical: Digital Logic Lab**

**1.Introduction to Xilinx software (VHDL)**

**Write the VHDL code for**

- 2. Realizing all logic gates.
- 3. Combinational Circuits
- 4. ADDER.
- 5. SUBTRACTOR.
- 6. MUX.
- 7. DE-MUX.
- 8. Encoder.
- 9. Decoder.
- 10. PAL.
- 11. PLA.

**Write theVHDL program for the following Sequential Logic Circuits**

- 12. Flip Flops.
- 13. Shift Registers.
- 14. Counters.
- 15. Memory Elements.

**PO-CO MAPPING TABLE**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
.CO 1	4	4	4	4	1	1	1	1	1	4	1	1
CO 2	4	4	4	4	1	1	1	1	1	4	1	1
.CO 3	4	4	4	4	1	1	1	1	1	4	1	1
CO 4	4	4	4	4	1	1	1	1	1	4	1	4

## CORE-3: Programming Using C++

---

### **Course Outcomes:**

**CO1:** understand the difference between structure-oriented programming and object-oriented programming.

**CO2:** apply various object-oriented features like class, object, constructor and destructor to solve various computing problems using C++ language.

**CO3:** understand and apply concepts of inheritance and operator-overloading,

**CO4:** Write programs that perform various operations on files

### **Unit-1**

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-2**

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, Dynamic Constructors, Destructors.

### **Unit-3**

Inheritance: Basics of Inheritance, Type of Inheritance, Virtual Base Classes, Abstract Classes, 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-4**

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

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

## Reference Books:

1. Bjarne Stroustrup, Programming - Principles and Practice using C++, 2/e, Addison-Wesley 2014
2. Herbtz Schildt, C++: The Complete reference, MGH, 4/ed.
3. P. C. Sethi, P. K. Behera, "Programming in C++"- Kalyani Publisher, Ludhiana

## CORE-3 Practical: Programming using C++ Lab

1. Write a Program to find greatest among three numbers using nested if...else statement.
2. Write a Program to check a number is prime or not.
3. Write a Program to find the GCD and LCM of two numbers.
4. Write a program to print the result for following series:  $1! + 2! + 3! + \dots$
5. Write a program to print multiplication table from 1 to 10.
6. Write a Program for Swapping of two numbers using pass by value.
7. Write a Program for Swapping of two numbers using pass by address.
8. Write a Program for Swapping of two numbers using pass by reference.
9. Write a Program to find sum of four numbers using default argument passing.
10. Write a Program to find square and cube of a number using inline function.
11. Write a Program to find the factorial of a number.
12. Write a Program to find reverse of a number.
13. Write a program to find sum of four numbers using default argument passing in member function.
14. Write a Program to find area of circle, triangle and rectangle using function overloading.
15. Write a program to distinguish the properties of static and non-static members.
16. Write a program to show the method of accessing static private member function.
17. Write a program to show the ways of calling constructors and destructors.
18. Write a program to perform ++ operator overloading using member function.
19. Write a program to perform ++ operator overloading using friend function.
20. Write a program to perform + operator overloading for two complex number addition.
21. Write a program to perform + operator overloading for string concatenation.
22. Write a program to perform single inheritance.
23. Write a program to perform multiple inheritance.
24. Write a program to create an integer array using new operator and find the sum and average of array elements.
25. Write a program to implement virtual destructor.
26. 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).
27. Write a program to Copy the contents of one file to other.

PO-CO MAPPING TABLE

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	4	1	1	1	1	1	1	1	1	1	2
CO2	2	1	5	5	1	1	1	1	1	1	5	2
CO3	2	1	5	5	1	1	1	1	1	1	5	2
CO4	2	4	4	1	1	1	1	1	1	1	1	2

### CORE-4: Data Structure

#### Course Outcomes:

**CO1:** Analyze performance of algorithms and implement various operations on array and Linked list.

**CO2:** Apply the basic operations of stacks and queues to solve real world problems.

**CO3:** Represent data using trees to use them in various real life applications.

**CO4:** Analyze and implement various sorting algorithms to solve real world problems

#### Unit-1

**Introduction:** Basic Terminology, Data structure, Time and space complexity, Review of Array, Structures, Pointers.

**Linked Lists:** Dynamic memory allocation, representation, Linked list insertion and deletion, Searching, Traversing in a list, Doubly linked list, Sparse matrices.

#### Unit-2

**Stack:** Definition, Representation, Stack operations, Applications (Infix-Prefix-Postfix Conversion & Evaluation, Recursion).

**Queues:** Definition, Representation, Types of queue, Queue operations, Applications.

#### Unit-3

**Trees:** Tree Terminologies, General Tree, Binary Tree, Representations, Traversing, BST, Operations on BST, Heap tree, AVL Search Trees, M-way search tree, Applications of all trees.

#### Unit-4

**Sorting:** Exchange sorts, Selection Sort, Bubble sort, Insertion Sorts, Merge Sort, Quick Sort, Radix Sort, Heap sort.

**Searching:** Linear search, Binary search.

### Text book

1. Classic Data Structure ,P.Samanta , PHI , 2/ed.

### REFERENCES

1. Ellis Horowitz,SartajSahni, “Fundamentals of Data Structures”, Galgotia Publications, 2000.
2. Sastry C.V., Nayak R, Ch. Rajaramesh, Data Structure & Algorithms, I.K.International Publishing House Pvt.Ltd, New Delhi.

### CORE – 4 Practical: Data Structure Lab

#### Write a C/ C++ Program for the followings

1. To insert and delete elements from appropriate position in an array.
2. To search an element and print the total time of occurrence in the array.
3. To delete all occurrence of an element in an array.
4. Array implementation of Stack.
5. Array implementation of Linear Queue.
6. Array implementation of Circular Queue.
7. To implement linear linked list and perform different operation such as node insert and delete, search of an item, reverse the list.
8. To implement circular linked list and perform different operation such as node insert and delete.
9. To implement double linked list and perform different operation such as node insert and delete.
10. Linked list implementation of Stack.
11. Linked list implementation of Queue.
12. Polynomial representation using linked list.
13. To implement a Binary Search Tree.
14. To represent a Sparse Matrix.
15. To perform binary search operation.
16. To perform Bubble sort.
17. To perform Selection sort.
18. To perform Insertion sort.
19. To perform Quick sort.
20. To perform Merge sort.

#### PO-CO MAPPING TABLE

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	5	2	4	2	2	1	1	1	1	1	1
CO2	5	5	2	4	2	2	1	1	1	1	1	1
CO3	5	5	2	4	2	2	1	1	1	1	1	1
CO4	5	5	2	4	2	2	1	1	1	1	1	1

## CORE – 5: Java Programming

---

### **Course Outcome**

**CO1:** understand basic concepts of OOP, introduction to classes and objects through Java Language and

apply

**CO2:** apply the concepts of constructors, overloading, parameter passing, access control, Inheritance

**CO3:** use Packages and Interfaces.

**CO4:** Implement Exception Handling, Threads and able to access and manipulate databases

### **Unit-1**

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 Commandline Argument, the Scanner class, BufferedReader class.

### **Unit-2**

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 constructor, 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-3**

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 Streams File 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-4**

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 Books:**

1. E. Balagurusamy, "Programming with Java", TMH, 4/Ed,

### **Reference books:**

1. Herbert Schildt, "The Complete Reference to Java", TMH, 10/Ed.

### **CORE – 5 Practical: Java Programming Lab**

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.
14. Write a program to demonstrate the concept of boxing and unboxing.
15. 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).
16. Write a program to create a multilevel package and also create 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.
17. Write a program that creates/illustrates different levels of protection in classes/subclasses belonging to same package or different packages
18. 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.
19. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
  20. 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).
  21. Write a program to demonstrate priorities among multiple threads.
  22. Write a program to demonstrate different mouse handling events like mouseClicked(), mouseEntered(), mouseExited(), mousePressed( ), mouseReleased() & mouseDragged().
  23. Write a program to demonstrate different keyboard handling events.

### PO-CO MAPPING TABLE

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	4	4		1	1	1	1	1	1	1	2
CO2	4	5	4	4	1	1	1	1	1	1	1	2
CO3	4	4	5	4	1	1	1	1	1	1	1	2
CO4	5		5	4	1	1	1	1	1	1	1	2

### CORE-6: Database Systems

#### **COURSE OUTCOMES (CO):**

- CO 1 Understand the basics of database management system
- CO 2 Use Structured Query Language (SQL) for database Creation and manipulation.
- CO 3 Demonstrate the working of different concepts of DBMS.
- CO 4 Construct a database by using data definition, data manipulation and control languages.
- CO 5 Implement and test the project developed for an application. Apply mathematical and formal techniques for solving problems in computer science related to database applications

#### **Unit-1**

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-2

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-3

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-4

Introduction to Transaction Processing Concepts and Theory: Introduction to Transaction Processing, 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 Book:

1. Fundamentals of Database Systems, 6th edition, RamezElmasri, ShamkantB.Navathe, Pearson Education

### Reference Book:

1. An Introduction to Database System, Date C.J.- Pearson Education, New Delhi- 2005

## CORE-6 Practical: Database Systems Labs

---

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

### Query List

1. Query to display Employee Name, Job, Hire Date, Employee Number; for each employee with the Employee Number appearing first.
2. Query to display unique Jobs from the Employee Table.
3. Query to display the Employee Name concatenated by a Job separated by a comma.
4. Query to display all the data from the Employee Table. Separate each Column by a comma and name the said column as THE\_OUTPUT.
5. Query to display the Employee Name and Salary of all the employees earning more than \$2850.
6. Query to display Employee Name and Department Number for the Employee No= 7900.
7. Query to display Employee Name and Salary for all employees whose salary is not in the range of \$1500 and \$2850.
8. Query to display Employee Name and Department No. of all the employees in Dept 10 and Dept 30 in the alphabetical order by name.
9. Query to display Name and Hire Date of every Employee who was hired in 1981.
10. Query to display Name and Job of all employees who don't have a current Manager.
11. Query to 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. Query to display Name of all the employees where the third letter of their name is 'A'.
14. Query to 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 Mangers Employee No = 7788.
15. Query to display Name, Salary and Commission for all employees whose Commission Amount is 14 greater than their Salary increased by 5%.
16. Query to display the Current Date.
17. Query to display Name, Hire Date and Salary Review Date which is the 1<sup>st</sup>Monday after six months of employment.
18. Query to display Name and calculate the number of months between today and the date each employee was hired.
19. Query to display the following for each employee <E-Name> earns < Salary> monthly but wants <3\*Current Salary>. Label the Column as Dream Salary.
20. Query to display Name with the 1<sup>st</sup>letter capitalized and all other letter lower case and length of their name of all the employees whose name starts with 'J', 'A' and 'M'.
21. Query to display Name, Hire Date and Day of the week on which the employee started.
22. Query to display Name, Department Name and Department No for all the employees.
23. Query to display Unique Listing of all Jobs that are in Department # 30.
24. Query to display Name, Department Name of all employees who have an 'A' in their name.
25. Query to display Name, Job, Department No. and Department Name for all the employees working at the Dallas location.
26. Query to display Name and Employee no. Along with their Manger's Name and the

- Manager's employee no; along with the Employees Name who do not have a Manager.
27. Query to 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. Query to display Name and Salaries represented by asterisks, where each asterisk (\*) signifies \$100.
  29. Query to display the Highest, Lowest, Sum and Average Salaries of all the employees.
  30. Query to display the number of employees performing the same Job type functions.
  31. Query to display the no. of managers without listing their names.
  32. Query to display the Department Name, Location Name, No. of Employees and the average salary for all employees in that department.
  33. Query to display Name and Hire Date for all employees in the same dept. as Blake.
  34. Query to display the Employee No. And Name for all employees who earn more than the average salary.
  35. Query to display Employee Number and Name for all employees who work in a department with any employee whose name contains a 'T'.
  36. Query to display the names and salaries of all employees who report to King.
  37. Query to display the department no, name and job for all employees in the Sales department.

CO-PO-MAPPING

CO \ PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	5	4	2	1	5	1	1	1	1	1	1	1
CO 2	4	4	4	4	5	1	1	1	1	1	1	1
CO 3	4	4	4	5	5	1	1	1	1	1	1	1
CO 4	5	4	5	1	5	1	1	1	1	1	1	1

---

**CORE – 7: Discrete Mathematical Structure**

---

**Course Outcomes:**

After competing this course, students will be able to:

**CO1.** Analyze statements using propositional and predicate logic, prove theorems using mathematical induction, understand sets, functions and relations and their properties.

**CO2.** Solve counting problems using counting principles, permutations, combinations, and pigeonhole principle, solve linear and non-linear recurrence relations using generating functions.

**CO3.** Apply principles and concepts of graph theory to solve real world problems.

**CO4.** Model DFAs, NFAs, grammars for different languages, minimize DFAs, apply pumping lemma to prove a language is not regular.

**Unit-1**

**Logics and Proof:** Propositional Logic, Propositional Equivalences, Predicates and Quantifiers  
Nested Quantifiers, Rules inference, Mathematical Induction.

**Sets and Functions:** Sets, Relations, Functions, Closures of Equivalence Relations, Partial  
ordering well ordering, Lattice, Sum of products and product of sums principle of Inclusions and  
Exclusions

## **Unit-2**

Combinatory: Permutations, Combinations, Pigeonhole principle

Recurrence Relation: Linear and Non-linear Recurrence Relations, Solving Recurrence Relation  
using Generating Functions.

## **Unit-3**

Graphs: Introduction to graphs, graphs terminologies, Representation of graphs, Isomorphism,  
Connectivity & Paths: Connectivity, Euler and Hamiltonian Paths, Introduction to tree, tree  
traversals, spanning tree and tree search: Breadth first search, Depth first search, cut-set, cut-  
vertex.

## **Unit-4**

**Modeling Computation:** Finite State Machine, Deterministic Finite Automata (DFA), Non-  
Deterministic Finite Automata (NFA), Grammars and Language, Application of Pumping Lemma  
for Regular Language.

### **Text Books:**

1. "Discrete Mathematics and its Applications with Combinatory and Graph Theory" 7<sup>th</sup>  
edition by Kenneth H. Rosen.

### **Reference Books:**

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

## **CORE – 7 Practical: Discrete Mathematical Structure Lab**

**Write the following programs using C/ C++**

1. Tower of Hanoi
2. Graph representation using Adjacency List.
3. Graph representation using Adjacency Matrix.
4. String Matching using finite state machine.
5. Detecting whether a number is even or odd using Finite State Machine.
6. To identify keywords such as char, const, continue using Finite State Machine.
7. To find the power set for a given set.
8. To find GCD of two numbers using recursion.
9. To find Binomial coefficients.
10. To find Permutation and Combination result for a given pair of values n and r.
11. To check a number is prime or not.
12. To calculate the Euclidean distance between two points.

13. To find the Roots of polynomials.  
 14. Find the shortest path pair in a plane.

**Mapping of Course Outcomes with the Program Outcomes:**

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	5	2	5	2	4	1	1	1	2	2	2
CO2	5	5	4	4	2	2	1	1	1	1	2	2
CO3	5	5	4	5	2	2	1	1	1	1	2	2
CO4	5	5	4	4	2	2	1	1	1	1	2	2

**CORE-8: Operating System**

---

**CO 1:** Able to understand the difference between different types of modern operating systems, virtual machines and their structure of implementation and applications.

**CO 2:** Able to understand the difference between process & thread, issues of scheduling of user-level processes / threads and their issues & use of locks, semaphores, monitors for synchronizing multiprogramming with multithreaded systems and implement them in multi threaded programs.

**CO 3:** Able to understand the concepts of deadlock in operating systems and how they can be managed / avoided and implement them in multiprogramming system.

**CO 4:** Able to understand the design and management concepts along with issues and challenges of main memory, virtual memory and file system.

**Unit-1**

Introduction to Operating System, System Structures: Operating system services, system calls, system programs, Operating system design and implementation, Operating system structure.

**Unit-2**

Process Management: Process Concept, Operations on processes, Process scheduling and algorithms, Inter-process Communication, Concepts on Thread and Process, Deadlocks: Deadlock detection, deadlock prevention, and deadlock avoidance fundamentals.

**Unit-3**

Memory Management Strategies: Swapping, Contiguous Memory Allocation, Paging, Segmentation, Virtual Memory Management: Concepts, implementation (Demand Paging), Page Replacement, Thrashing.

## **Unit-4**

**Storage Management: File System concept, Access Methods, File System Mounting, File Sharing and File Protection, Implementing File Systems, Kernel I/O Systems.**

**Text book** – Operating System Concepts, Abraham Silberschatz, Peter B. Galvin, and Greg Gagne, Eighth Edition, Wiley Student Edition 2009.

### **Reference book:**

1. Modern Operating System ,Tanenbaum ,Pearson , 4/ed. 2014
2. Richard F Ashley, Linux with Operating System Concepts, Chapman and Hall/CRC  
Published August 26, 2014
3. Richard Blum, Linux Command Line and Shell Scripting Bible, O' Reilly

### **CORE-8 Practical: Operating System Lab**

1. Write a program (using *fork()* and/or *exec()* commands) where parent and child execute:
  - a) same program, same code.
  - b) same program, different code.
  - c) before terminating, the parent waits for the child to finish its task.
2. Write a program to report behavior of Linux kernel including kernel version, CPU type and model. (CPU information)
3. Write a program to report behavior of Linux kernel including information on configured memory, amount of free and used memory. (memory information)
4. Write a program to print file details including owner access permissions, file access time, where file name is given as argument.
5. Write a program to copy files using system calls.
6. Write a program using C to implement FCFS scheduling algorithm.
7. Write a program using C to implement Round Robin scheduling algorithm.
8. Write a program using C to implement SJF scheduling algorithm.
9. Write a program using C to implement non-preemptive priority based scheduling algorithm.
10. Write a program using C to implement preemptive priority based scheduling algorithm.
11. Write a program using C to implement SRTF scheduling algorithm.
12. Write a program using C to implement first-fit, best-fit and worst-fit allocation strategies.



## PO-CO MAPPING

POs \ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
	5	5	4	4	4	1	1	1	1	4	4	1
	5	5	5	5	5	1	4	1	4	4	4	1
	4	5	5	5	5	1	1	1	4	5	5	1
	5	5	5	5	5	5	1	1	4	4	5	1

## CORE – 9: Computer Networks

---

### Course OutcomeUnit

**CO1:**To understand various types of signals, transmissions, multiplexing and networks.

**CO2:**To understand about error detection and error correction techniques.

**CO3:**To learn about IPv4 and IPv6 and various transport layer protocols.

**CO4:**To learn about Email and protocols used to transfer data.

### **UNIT-1**

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-2**

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-3**

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-4

Network Layer: Logical Addressing, IPv4 Addresses, IPv6 Addresses, Virtual-Circuit Networks: Frame Relay and ATM, Transport Layer: Process-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 Books:

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

### Reference Books:

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

## CORE – 9 Practical:Computer Networks Lab

### Use C/C++/ any Network Simulator

1. Simulate Even Parity generator and checker.
2. Simulate two dimensional Parity generator and checker.
3. Simulate checksum generator and checker.
4. Simulate Hamming code method.
5. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
6. Simulate and implement stop and wait protocol for noisy channel.
7. Simulate and implement go back n sliding window protocol.
8. Simulate and implement selective repeat sliding window protocol.
9. Simulate and implement distance vector routing algorithm.

### PO-CO MAPPING TABLE

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	5	4	4	4	2	1	1	1	1	1	1
CO2	5	5	4	4	4	2	1	1	1	1	1	1
CO3	5	5	4	4	4	2	1	1	1	1	1	1
CO4	5	5	4	4	4	2	1	1	1	1	1	1

**COURSE OUTCOMES (CO):**

After completion of the course, a student will be able to

CO 1 Analyze the background processes involved in computer graphics displays, understanding of Algorithms.

CO 2 Use Mathematics in Vector. Create Segments and apply clipping to different shapes.

CO 3 Understand and apply algorithms used in Computer Graphics.

CO 4 Apply methods suitable for 2D and 3D Transformation such as Translation, Rotation, Scaling, Reflection, Shear etc.

CC 5 Apply Clipping algorithm for viewing transformation

**Unit-1**

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-2**

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

**Unit-3**

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-4**

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**

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

**Reference books**

1. D. Hearn and M. Baker, "Computer Graphics with Open GL", Pearson, 2/ed.
2. D. F. Rogers, "Procedural Elements for Computer Graphics", MGH

## **CORE – 10 Practical: Computer Graphics Lab**

### **Develop the programs using C/C++ or Java**

1. Write a program to implement Brenham'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 transformations on a 2D object (use homogenous coordinates).
7. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.

CO-PO-PSOMAPPING

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	5	2	1	1	2	2	1	1	4	2	5	1
CO2	2	5	2	4	2		1	1	5	4	2	1
CO3	4	5	1	4	2	2	1	2	4	5	5	1
CO4	2	4	4	2		4	2	1	5	5	4	1

## **CORE – 11: Web Technologies**

---

### **Course Outcome**

**CO1:** Develop simple web pages using HTML, and Cascading Styles sheets

**CO2:** Develop web pages using DHTML and Cascading Styles sheets.

**CO3:** Develop a dynamic web pages using JavaScript (client side programming).

**CO4:** Develop an interactive web applications using PHP.

### **Unit-1**

**Web Essentials: Clients, Servers and Communication:**

**The Internet –Basic Internet protocols–The WWW,HTTP request message –response message,web clients web servers – case study.**

**Introduction to HTML: HTML, HTML domains, basic structure of an HTML document–creating an HTML document, mark up tags, heading, paragraphs, line breaks, HTML tags. Elements of HTML, working with text, lists, tables and frames, working with hyperlink, images and multimedia, forms and controls**

## **Unit-2**

Introduction to cascading style sheets: Concepts of CSS, creating style sheet, CSS properties, CSS styling(background, text format, controlling fonts), working with the block elements and objects. Working with lists and tables, CSS ID and class.Box model(introduction, border properties, padding properties, margin properties), CSS colour, grouping, Dimensions, display, positioning, floating, align, pseudo class, Navigation bar, image sprites.

## **Unit-3**

JavaScript: Client side scripting, what is JavaScript, simple JavaScript, variables, functions, conditions, loops and repetitions. JavaScript and objects, JavaScript own objects, the DOM and web browser environment, forms and validations.

DHTML: Combining HTML, CSS, JavaScript, events and buttons, controlling your browser.

## **Unit-4**

PHP: Starting to script on server side, PHP basics, variables, data types, operators, expressions, constants, decisions and loop making decisions. Strings – creating, accessing strings, searching, replacing and formatting strings.Arrays: Creation, accessing array, multidimensional arrays, PHP with Database.

### **Text Book:**

1. Web Technologies – Black Book – DreamTech Press
2. Matt Doyle, Beginning PHP 5.3 (wrox-Wiley publishing)
3. John Duckett, Beginning HTML, XHTML, CSS and JavaScript.

### **Reference Book:**

3. HTML, XHTML and CSS Bible, 5ed, Wiley India-Steven M. Schafer.

## **CORE – 11 Practical: Web Technology Lab**

1. Acquaintance with elements, tags and basic structure of HTML files.
2. Practicing basic and advanced text for formatting.
3. Practice use of image, video and sound in HTML documents.
4. Designing of web pages- Document layout, list, tables.
5. Practicing Hyperlink of web pages, working with frames.
6. Working with forms and controls.
7. Acquaintance with creating style sheet, CSS properties and styling.
8. Working with background, text, font, list properties.
9. Working with HTML elements box properties in CSS.

10. Develop simple calculator for addition, subtraction, multiplication and division operation using java script.
11. Create HTML page with java script which takes integer number as a input and tells whether the number is odd or even.
12. Create HTML page that contains form with fields name, Email, mobile number, gender, favorite colour and button; now write a java script code to validate each entry. Also write a code to combine and display the information in text box when button is clicked.
13. Write a PHP program to check if number is prime or not.
14. Write a PHP program to print first ten Fibonacci numbers.
15. Create a MySQL data base and connect with PHP.
16. Write PHP script for string and retrieving user information from my SQL table.
  - a. Write a HTML page which takes Name, Address, Email and Mobile number from user (register PHP).
  - b. Store this data in MySQL data base.
  - c. Next page display all user in HTML table using PHP (display .PHP).
17. Using HTML, CSS, Javascript, PHP, MySQL, design a authentication module of a web page.

#### PO-CO MAPPING TABLE

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	4	5	1	1	1	1	1	5	4	4	2
CO2	4	4	5	1	4	1	1	1	5	4	4	2
CO3	4	4	5	1	1	1	1	1	5	4	4	4
CO4	4	4	5	1	1	1	1	1	5	4	4	4

#### **CORE – 12: Software Engineering**

CO 1: Ability to gather and specify requirements of the software projects.

CO 2: Ability to analyze software requirements with existing tools

CO 3: Able to understand and apply the basic project management practices in real life projects

CO 4: Able to differentiate different testing methodologies

#### **Unit-1**

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-2**

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-3

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-4

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:

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

#### Reference Books:

1. Software Engineering– Ian Somerville, 10/Ed, Pearson.
2. Software Engineering Concepts and Practice – UgrasenSuman, Cengage Learning India Pvt, Ltd.
3. R. Misra, C. Panigrahi, B. Panda: Principles of Software Engineering & System Design, YesDee Publication

### CORE – 12 Practical: Software Engineering Lab

#### S. No. Practical Title

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

**Sample 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

PO-CO MAPPING TABLE

POs COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	4	1	1	4	4	1	1	4	5	4	4
CO 2	4	5	4	4	5	1	1	1	1	4	1	1
CO 3	5	5	4	5	5	1	1	1	4	4	1	5
CO 4	4	5	5	5	5	1	1	1	4	4	1	1



## CORE–13: Artificial Intelligence

---

### Course Outcomes

- CO 1: To learn the basic concepts of AI principles and approaches.
- CO 2: To develop the basic understanding of the building blocks of AI.
- CO 3: To learn how to represent Knowledge
- CO 4: To learn basic concept of Natural Language Processing

### **Unit-1**

Introduction to Artificial Intelligence, Background and Applications, Turing Test and Rational Agent approaches to AI, Introduction to Intelligent Agents, their structure, behavior and environment.

### **Unit-2**

Problem Solving and Searching Techniques: Problem Characteristics, Production Systems, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristics Search Techniques: Best First Search, A\* algorithm, Constraint Satisfaction Problem, Introduction to Game Playing, Min-Max and Alpha-Beta pruning algorithms.

### **Unit-3**

Knowledge Representation : Introduction to First Order Predicate Logic, Resolution Principle, Unification, Semantic Nets, Conceptual Dependencies, Frames, and Scripts, Production Rules, Conceptual Graphs.

### **Unit-4**

Dealing with Uncertainty and Inconsistencies Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference, Possible World Representations, Basics of NLP.

### **Text books**

1. Artificial Intelligence a Modern Approach, Stuart Russell and Peter Norvig, Pearson 3/ed.

### **Reference books**

1. Artificial Intelligence, Rich & Knight , TMG , 3 e/d.
2. DAN.W. Patterson, Introduction to A.I and Expert Systems – PHI, 2007
3. W.F. Clocksin and Mellish, Programming in PROLOG, Narosa Publishing House, 3rd edition, 2001

## CORE–13 Practical: Artificial Intelligence Lab

### Write a Prolog program

1. To find the factorial of a number
2. To remove the nth item from a list.
3. To find the permutation of a set.
4. To implement append for two lists.

5. To implement palindrome.
6. To find the greater of two numbers X and Y.
7. To find the greatest number in the list of numbers.
8. To find the sum of given list of numbers.
9. To find the reverse of a list.
10. To solve 8 queens problem.
11. To solve 8-puzzle problem using best first search
12. To implement DFS.
13. To implement BFS.
14. To implement best first search.
15. To solve traveling salesman problem.

PO-CO MAPPING TABLE

POs COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	4	1	1	4	4	1	1	4	5	4	4
CO 2	4	5	4	4	5	1	1	1	1	4	1	1
CO 3	5	5	4	5	5	1	1	1	4	4	1	5
CO 4	4	5	5	5	5	1	1	1	4	4	1	1

### CORE – 14: Algorithm Design Techniques

#### Course Outcomes:

After completing this course, students will be able to:

**CO1.** Write sorting algorithms, analyze the efficiency of algorithms using asymptotic notations, argue the correctness of algorithms using loop invariants.

**CO2.** Understand the concept of hashing, describe and apply the divide-and-conquer paradigm, derive and solve recurrences describing the performance of divide-and-conquer

**CO3.** Synthesize greedy and dynamic-programming algorithms, solve and analyze several problems using greedy and dynamic programming techniques.

**CO4.** Explain major graph algorithms and analyze the time complexity

#### Unit-1

**Introduction:** Algorithm specification: Pseudo code, 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-2**

**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-3**

**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.

## **Unit-4**

**Graph Algorithms:** Representations of Graphs, Breadth-first search, Depth-first search, Topological sort, Minimum Spanning Trees: Prim's and Kruskal's algorithm, Single-source shortest paths: Bellman-Ford algorithm, Dijkstra's algorithm.

### **Text books**

1. Introduction to Algorithms, by Thomas H, Cormen, Charles E. Leiserson , Ronald L. Rivest, Clifford Stein, PHI.

### **Reference books**

1. Algorithm Design, by Jon Kleinberg, Eva Tardos.

## **CORE – 14 Practical: Algorithm Design Techniques Lab**

**Using C or C++ implement the following**

1. Quick sort.
2. Heap sort.
3. Merge sort.
4. Matrix Multiplication using recursion.
5. Linear Search.
6. Binary Search.
7. Huffman code.
8. Fractional knapsack problem.
9. Matrix chain multiplication.
10. Longest Common Subsequence.
11. Prim's algorithm.
12. Kruskal's algorithm.
13. BFS.
14. DFS.
15. Dijkstra Algorithm.

### Mapping of Course Outcomes with the Program Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	5	5	2	2	2	1	1	1	2	1	2
CO2	5	5	5	4	4	2	1	1	1	2	1	2
CO3	5	5	5	2	4	2	1	1	1	2	1	2
CO4	5	4	5	2	4	2	1	1	1	2	1	2

### DSE-1: Numerical Techniques

---

#### **Course Outcomes:**

After completing this course, students will be able to:

**CO1.** Understand computer arithmetic and truncation errors in detail.

**CO2.** Find numerical techniques to find the roots of algebraic equations, and check the accuracy of the solutions.

**CO3.** Describe various interpolating methods and apply several numerical methods in real life problems.

**CO4.** Apply numerical methods to find numerical integration, and numerical solutions of ordinary differential equations.

#### **Unit-1**

Floating point representation and computer arithmetic, Significant digits, Errors: Round-off error, Local truncation error, Global truncation error, Order of a method, Convergence and terminal conditions, Efficient computations.

#### **Unit-2**

Bisection method, Secant method, Regula-Falsi method Newton-Raphson method, Newton's method for solving nonlinear systems.

#### **Unit-3**

Interpolation: Lagrange's form and Newton's form Finite difference operators, Gregory Newton forward and backward differences Interpolation Piecewise polynomial interpolation: Linear interpolation.

## Unit-4

Numerical integration: Trapezoid rule, Simpson's rule (only method), Newton–Cotes formulas, Gaussian quadrature, Ordinary differential equation: Euler's method Modified Euler's methods, Runge-Kutta second methods

### Text books

1. S.S. Sastry, "Introductory Methods of Numerical Analysis", EEE , 5/ed.
2. M.K. Jain, S.R.K. Iyengar and R.K. Jain, Numerical Methods for Scientific and Engineering Computation, New Age International Publisher, 6/e (2012)

### Reference books

1. Numerical Analysis: J.K. Mantri & S. Prahan, Laxmi Publication.
2. Introduction to Numerical Analysis, Josef Stoer and Roland Bullrich, Springer.

### DSE – 1 Practical: Numerical Techniques Lab

#### Implement using C/ C++ or MatLab

1. Find the roots of the equation by bisection method.
2. Find the roots of the equation by secant/Regula–False method.
3. Find the roots of the equation by Newton's method.
4. Find the solution of a system of nonlinear equation using Newton's method.
5. Find the solution of tri-diagonal system using Gauss Thomas method.
6. Find the solution of system of equations using Jacobi/Gauss-Seidel method.
7. Find the cubic spline interpolating function.
8. Evaluate the approximate value of finite integrals using Gaussian/Romberg integration.
9. Solve the boundary value problem using finite difference method.

#### Mapping of Course Outcomes with the Program Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	4	2	4	2	1	1	1	1	2	1	2
CO2	5	4	4	4	5	1	1	1	1	2	1	2
CO3	5	4	4	4	5	1	1	1	1	2	1	2
CO4	5	4	4	4	5	1	1	1	1	2	1	2

### Course outcomes

**CO1:**To learn the basics of UNIX OS, UNIX commands and File system.

**CO2:**To familiarize students with the Linux environment.

**CO3:** To learn fundamentals of shell scripts and shell programming.

**CO4:**To be able to write simple programs using UNIX.

### **Unit-1**

**Introduction:** Unix Operating systems, Difference between Unix and other operating systems, Features and Architecture, Installation, Booting and shutdown process, System processes (an overview), External and internal commands, Creation of partitions in OS, Processes and its creation phases – Fork, Exec, wait, exit.

### **Unit-2**

**User Management and the File System:** Types of Users, Creating users, Granting rights, User management commands, File quota and various file systems available, File System Management and Layout, File permissions, Login process, Managing Disk Quotas, Links (hard links, symbolic links)

### **Unit-3**

**Shell introduction and Shell Scripting:** Shell and various type of shell, Various editors present in Unix, Different modes of operation in vi editor, Shell script, Writing and executing the shell script, Shell variable (user defined and system variables), System calls, Using system calls, Pipes and Filters.

### **Unit-4**

**Unix Control Structures and Utilities:** Decision making in Shell Scripts (If else, switch), Loops in shell, Functions, Utility programs (cut, paste, join, tr, uniq utilities), Pattern matching utility (grep).

### **Text Books:**

1. Sumitabha, Das, Unix Concepts And Applications, Tata McGraw-Hill Education, 2017, 4/Ed.

### **Reference Books:**

1. Nemeth Synder& Hein, Linux Administration Handbook, Pearson Education,2010, 2/ Ed.

## DSE – 2 Practical: Unix Programming Lab

1. Write a shell script to check if the number entered at the command line is prime or not.
2. Write a shell script to modify “cal” command to display calendars of the specified months.
3. Write a shell script to modify “cal” command to display calendars of the specified range of months.
4. Write a shell script to accept a login name. If not a valid login name display message “Entered login name is invalid”.
5. Write a shell script to display date in the mm/dd/yy format.
6. Write a shell script to display on the screen sorted output of “who” command along with the total number of users.
7. Write a shell script to display the multiplication table of any number.
8. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
9. Write a shell script to find the sum of digits of a given number.
10. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
11. Write a shell script to find the LCD(least common divisor) of two numbers.
12. Write a shell script to perform the tasks of basic calculator.
13. Write a shell script to find the power of a given number.
14. Write a shell script to find the greatest number among the three numbers.
15. Write a shell script to find the factorial of a given number.
16. Write a shell script to check whether the number is Armstrong or not.

**PO-CO MAPPING TABLE**

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	5	4	4	4	2	1	1	1	1	1	1
CO2	5	5	4	4	4	2	1	1	1	1	1	1
CO3	5	5	4	4	4	2	1	1	1	1	1	1
CO4	5	5	4	4	4	2	1	1	1	1	1	1

## **DSE-3: Data Science**

---

### Course Outcomes

**CO 1:** Able to do some innovative work with applying the knowledge gained from various courses undergone in the earlier years.

**CO 2:** Able to know the complete project life cycle and the project time estimation & its management.

**CO 3:** Able to gain knowledge of various simulation tools.

**CO 4:** Able to culture working in a team

## Unit-1

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

## Unit-2

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

## Unit-3

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

## Unit-4

**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 Books

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

### Reference Books

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

### DSE-3 Practical: Data Science Lab

1. Write a program that prints "Hello World" to the screen.
2. Write a program that asks the user for a number n and prints the sum of the numbers 1 to n
3. Write a program that prints a multiplication table for numbers up to 12.
4. Write a function that returns the largest element in a list.
5. Write a function that computes the running total of a list.
6. Write a function that tests whether a string is a palindrome.
7. Implement linear search.
8. Implement binary search.
9. Implement matrices addition, subtraction and Multiplication
10. Fifteen students were enrolled in a course. Their ages were:

20 20 20 20 20 21 21 21 22 22 22 22 23 23 23



- i. Find the median age of all students under 22 years.
- ii. Find the median age of all students.
- iii. Find the mean age of all students.
- iv. Find the modal age for all students.
- v. Two more students enter the class. The age of both students is 23. What is now mean, mode and median?

POs COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO 1	5	5	5	4	1	1	1	1	1	4	1	5
CO 2	4	5	4	4	4	1	1	1	1	1	1	5
CO 3	4	4	5	5	5	1	1	1	1	4	1	1
CO 4	2	4	5	4	5	1	1	1	5	4	1	5

**DSE-4:**  
**PROJECT WORK/ DISSERTATION OR DATA MINING**

**Course Outcomes**

**CO 1:** Able to do some innovative work with applying the knowledge gained from various courses undergone in the earlier years.

**CO 2:** Able to know the complete project life cycle and the project time estimation & its management.

**CO 3:** Able to gain knowledge of various simulation tools.

**CO 4:** Able to culture working in a team

**Unit-1**

**Data Warehouse Fundamentals:** Introduction to Data Warehouse, OLTP Systems, OLAP, Differences between OLTP and OLAP, Characteristics of Data Warehouse, Functionality of Data Warehouse, Advantages and Applications of Data Warehouse, Advantages, Applications, Top- Down and Bottom-Up Development Methodology, Tools for Data warehouse development, Data Warehouse Types, Data cubes

**Unit-2**

**Introduction to Data Mining:** Data mining, Functionalities, Data Preprocessing: Preprocessing the Data, Data cleaning, Data Integration and Transformation, Data reduction, Discretization and Concept hierarchies.

### Unit-3

**Mining Association Rules:** Basics Concepts – Single Dimensional Boolean Association Rules from Transaction Databases, Multilevel Association Rules from transaction databases, Multi dimension Association Rules from Relational Database and Data Warehouses, Apriori Algorithm, FP-Tree algorithm

### Unit-4

**Classification and Prediction:** Introduction, Issues, Decision Tree Induction, Naïve Bayesian Classification, Classification based on Concepts from Association Rule Mining, Classifier Accuracy.

#### Text Books:

1. J.Han and M. Kamber, Data Mining Concepts and Techniques, Elsevier, 2011

#### Reference Books:

1. K.P. Soman ,ShyamDiwakar, V.Ajay ,2006, Insight into Data Mining Theory and Practice, Prentice Hall of India Pvt. Ltd - New Delhi.
2. Data Mining Techniques, Arun K. Pujari, Universities Press, 2006
3. Modern Approaches of Data Mining: Theory & Practice, M. Panda, S. Dehuri, M. R. Patra, Narosa Publishing House, 2018.

### **DSE – 4 Practical: Data Mining Lab** **Using Sci. Lab/ Mat. Lab/ C/ Python/ R**

1. Build a Data Warehouse and perform it's operations.
2. Perform data preprocessing tasks and Demonstrate performing association rule mining on data sets.
3. Demonstrate performing classification on data sets.
4. Demonstrate performing clustering on data sets.
5. Demonstrate performing Regression on data sets.
6. Credit Risk Assessment. Sample Programs using German Credit Data.
7. Sample Programs using Hospital Management System.

**PO-CO MAPPING TABLE**

POs \ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO 1	5	5	5	4	1	1	1	1	1	4	1	5
CO 2	4	5	4	4	4	1	1	1	1	1	1	5
CO 3	4	4	5	5	5	1	1	1	1	4	1	1
CO 4	2	4	5	4	5	1	1	1	5	4	1	5

## GE-1: Computer Fundamentals

---

### **Course Outcomes:**

**CO1:** understand about the definition and data representation of computer.

**CO2:** understand different devices and memory of computer.

**CO3:** understand and apply concepts of computer organization and architecture,

**CO4:** understand some recent emerging technologies and their applications.

### **Unit-1**

**Introduction:** Introduction to computer system, uses, types.

**Data Representation:** Number systems and character representation, binary arithmetic

**Human Computer Interface:** Types of software, Operating system as user interface, utility programs

### **Unit-2**

**Devices:** Input and output devices (with connections and practical demo), keyboard, mouse, joystick, scanner, OCR, OMR, bar code reader, web camera, monitor, printer, plotter

**Memory:** Primary, secondary, auxiliary memory, RAM, ROM, cache memory, hard disks, optical disks

### **Unit-3**

**Computer Organisation and Architecture:** C.P.U., registers, system bus, main memory unit, cache memory, Inside a computer, SMPS, Motherboard, Ports and Interfaces, expansion cards, ribbon cables, memory chips, processors.

### **Unit-4**

**Overview of Emerging Technologies:** Bluetooth, cloud computing, big data, data mining, mobile computing and embedded systems.

### **Text Books:**

1. A. Goel, Computer Fundamentals, Pearson Education, 2010.

### **Reference Books:**

1. P. Aksoy, L. DeNardis, Introduction to Information Technology, Cengage Learning, 2006

2. P. K.Sinha, P. Sinha, Fundamentals of Computers, BPB Publishers, 2007

## GE-1 Practical: Computer Fundamentals Lab

Practical exercises based on MS Office tools including document preparation and spreadsheet handling packages.

### **MS Word:**

1. Prepare a grocery list having four columns (Serial number, The name of the product, quantity and price) for the month of February, 2019.

- Font specifications for Title (Grocery List): 14-point Arial font in bold and italics.
- The headings of the columns should be in 12-point and bold.
- The rest of the document should be in 10-point Times New Roman.
- Leave a gap of 12-points after the title.

2. Create a telephone directory.

- The heading should be 16-point Arial Font in bold
- The rest of the document should use 10-point font size
- Other headings should use 10-point Courier New Font.
- The footer should show the page number as well as the date last updated.

3. Design a time-table form for your college.

- The first line should mention the name of the college in 16-point Arial Font and should be bold.
- The second line should give the course name/teacher's name and the department in 14-point Arial.

4. Create the following documents:

- a) A newsletter with a headline and 2 columns in portrait orientation, including at least one image surrounded by text.
- b) Use a newsletter format to promote upcoming projects or events in your classroom or college.

5. Enter the following data into a table given below:

Salesperson	Dolls	Trucks	Puzzles
Kennedy, Sally	1327	1423	1193
White, Pete	1421	3863	2934
Pillar, James	5214	3247	5467
York, George	2190	1278	1928
Banks, Jennifer	1201	2528	1203
Atwater, Kelly	4098	3079	2067

Add a column Region (values: S, N, N,S,S,S) between the Salesperson and Dolls columns to the given table Sort your table data by Region and within Region by Salesperson in ascending order: In this exercise, you will add a new row to your table, place the word "Total" at the bottom of the Salesperson column, and sum the Dolls, Trucks, and Puzzles columns.

**MS Excel**

6. Given the following worksheet

	A	B	C	D
1	Roll No.	Name	Marks	Grade
2	1001	Sachin	99	
3	1002	Sehwag	65	
4	1003	Rahul	41	
5	1004	Sourav	89	
6	1005	Har Bhajan	56	

Calculate the grade of these students on the basis of following guidelines:

If Marks	Then Grade
$\geq 80$	A+
$\geq 60 < 80$	A
$\geq 50 < 60$	B
$< 50$	F

7. Given the following worksheet

	A	B	C	D	E	F	G	
1	Salesman	Sales in (Rs.)						
2	No.	Qtr1	Qtr2	Qtr3	Qtr4	Total	Commission	
3	S001	5000	8500	12000	9000			
4	S002	7000	4000	7500	11000			
5	S003	4000	9000	6500	8200			
6	S004	5500	6900	4500	10500			
7	S005	7400	8500	9200	8300			
8	S006	5300	7600	9800	6100			

Calculate the commission earned by the salesmen on the basis of following Candidates:

If Total Sales	Commission
$< 20000$	0% of sales
$> 20000$ and $< 25000$	4% of sales
$> 25000$ and $< 30000$	5.5% of sales
$> 30000$ and $< 35000$	8% of sales
$\geq 35000$	11% of sales

The total sales is sum of sales of all the four quarters.

8.

Create Payment Table for a fixed Principal amount, variable rate of interests and time in the format below:

No. of Instalments	5%	6%	7%	8%	9%
3	XX	XX	XX	XX	XX
4	XX	XX	XX	XX	XX
5	XX	XX	XX	XX	XX
6	XX	XX	XX	XX	XX

9. A company XYZ Ltd. pays a monthly salary to its employees which consists of basic salary, allowances & deductions. The details of allowances and deductions are as follows:

**Allowances**

- HRA Dependent on Basic
  - 30% of Basic if Basic  $\leq$ 1000
  - 25% of Basic if Basic  $>$ 1000 & Basic  $\leq$ 3000
  - 20% of Basic if Basic  $>$ 3000
- DA Fixed for all employees, 30% of Basic
- Conveyance Allowance Rs. 50/- if Basic is  $\leq$ 1000  
Rs. 75/- if Basic  $>$ 1000 & Basic  $\leq$ 2000  
Rs. 100 if Basic  $>$ 2000
- Entertainment Allowance NIL if Basic is  $\leq$ 1000  
Rs. 100/- if Basic  $>$  1000

**Deductions**

- Provident Fund 6% of Basic
- Group Insurance Premium Rs. 40/- if Basic is  $\leq$ 1500  
Rs. 60/- if Basic  $>$  1500 & Basic  $\leq$ 3000  
Rs. 80/- if Basic  $>$ 3000

Calculate the following:

Gross Salary = Basic + HRA + DA + Conveyance + Entertainment

Total deduction = Provident Fund + Group Insurance Premium

Net Salary = Gross Salary – Total Deduction

9.

The following table gives year wise sale figure of five salesmen in Rs.

Salesman	2000	2001	2002	2003
S1	10000	12000	20000	50000
S2	15000	18000	50000	60000
S3	20000	22000	70000	70000
S4	30000	30000	100000	80000
S5	40000	45000	125000	90000

- (a) Calculate total sale year wise.
- (b) Calculate the net sale made by each salesman
- (c) Calculate the maximum sale made by the salesman
- (d) Calculate the commission for each salesman under the condition.
  - (i) If total sales  $>$ 4,00,000 give 5% commission on total sale made by the salesman.
  - (ii) Otherwise give 2% commission.
- (e) Draw a bar graph representing the sale made by each salesman.
- (f) Draw a pie graph representing the sale made by salesman in 2000.

**CO-PO MAPPING TABLE**

PO \ CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	5	1	1	1	1	1	1	1	1	1	1
CO2	5	2	1	1	1	1	1	1	1	1	1	1
CO3	5	2	1	1	1	1	1	1	1	1	1	1

CO4	5	2	1	1	1	1	1	1	1	1	1	1
-----	---	---	---	---	---	---	---	---	---	---	---	---

## GE – 2: C and Data Structure

---

### Course Outcome

**CO1:** Understand and formulate the algorithms to programs (in C language) and develop programs using the basic elements like control statements.

**CO2:** Apply modular programming approach and recursion mechanism to solve complex problems.

**CO3:** Implement Programs with pointers, and learn to use the pre-processors.

**CO4:** Apply the basic operations of stacks and queues and implement various sorting algorithms to solve real world problems.

### **Unit-1**

Algorithm, flowchart, program development steps, structure of C program, A Simple C program, identifiers, basic data types and sizes, Constants, variables, arithmetic, relational and logical operators, increment and decrement operators, conditional operator, bit-wise operators, assignment operators, expressions, type conversions, conditional expressions, precedence and order of evaluation.

Input-output statements, statements and blocks, if and switch statements, loops- while, do-while and for statements, break, continue, goto and labels, programming examples.

### **Unit-2**

Designing structured programs, Functions, basics, parameter passing, storage classes- extern, auto, register, static, scope rules, block structure, user defined functions, standard library functions, recursive functions, header files, C preprocessor, example c programs.

### **Unit-3**

Arrays- concepts, declaration, definition, accessing elements, storing elements, arrays and functions, two-dimensional and multi-dimensional arrays, applications of arrays. pointers- concepts, initialization of pointer variables, pointers and function arguments, address arithmetic, Character pointers and functions, pointers to pointers, pointers and multidimensional arrays, dynamic memory managements functions, command line arguments, C program examples.

### **Unit-4**

Introduction to data structures, representing stacks and queues in C using arrays, infix to post fix conversion, postfix expression evaluation, Applications of Queue.

Searching - Linear and binary search methods, sorting - Bubble sort, selection sort, Insertion sort, Quick sort.

**Text Books:**

1. E. Balagurusamy, "Programming in ANSI C", 4/e, (TMH)
2. Seymour Lipschutz, "Data Structure with C", - Schaum's Outlines MGH.

**Reference Books:**

1. B. Kernighan & Dennis Ritchie, "The C Programming Language", 2/e PHI
2. P.C. Sethi, P.K. Behera, "Programming using C", Kalyani Publisher, Ludhiana
3. DataStructures Using C - A.S.Tanenbaum, Y. Langsam, M.J. Augenstein, PHI/Pearson.

**GE – 2 Practical: C and Data Structure Lab**

1. Write a Program to find the greatest among three numbers.
2. Write a Program to check a number is leap year or not.
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 function to find whether a given no. is prime or not.
7. Write a Program to compute factorial of a number.
8. Write a Program to print a triangle of stars as follows (take number of lines from user):
9. 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.
10. To insert and delete elements from appropriate position in an array.
11. To search an element and print the total time of occurrence in the array.
12. Array implementation of Stack.
13. Array implementation of Queue.
14. To perform Bubble sort.
15. To perform Selection sort.



# DEPARTMENT OF COMPUTER SCIENCE

## SYLLABUS OF PG PROGRAMME (M.Sc.)



**RAMA DEVI WOMEN'S UNIVERSITY**  
Vidya Vihar, Bhubaneswar-751022, Odisha  
Website: <https://rdwu.ac.in>

# RAMA DEVI WOMEN'S UNIVERSITY

## Syllabus for Masters in Science, Computer Science (2-Years Programme)



P. G. DEPARTMENT OF COMPUTER SCIENCE  
RAMA DEVI WOMEN'S UNIVERSITY  
VIDYA VIHAR, BHUBANESWAR-751022

2022-23

*Master*  
21.10.23  
Controller of Examinations  
R.D. Women's University  
Bhubaneswar

**DEPARTMENT OF COMPUTER SCIENCE (W.E.F.2022-23)**

**RAMA DEVI WOMEN'S UNIVERSITY**

**PG SYLLABUS STRUCTURE**

Semester-I								
Sl. No.	Nature of Course	Course Code	Paper Title	Units	Credits	Marks		
						Mid-Sem.	End-Sem.	Total
1	Hard Core	HC-101	Discrete Mathematical Structure	5	5	30	70	100
2	Hard Core	HC-102	Computer System Architecture	5	5	30	70	100
3	Hard Core	HC-103	Database Systems Implementation	5	5	30	70	100
4	Hard Core	HC-104	a) Database Systems Practical b) Python Practical	--	5	30	70	100
5	Allied Core	AC-101	Computer Applications in Teaching Learning (Course to be offered by e-learning center)	3	3	Mid-Sem 10+ Practical 10=20 marks	30	50
<b>Total</b>					<b>23</b>	<b>140</b>	<b>310</b>	<b>450</b>

Semester-II								
Sl. No.	Nature of Course	Course Code	Paper Title	Units	Credits	Marks		
						Mid-Sem.	End-Sem.	Total
6	Hard Core	HC-201	Data structure & Algorithm	5	5	30	70	100
7	Hard Core	HC-202	Operating Systems	5	5	30	70	100
8	Hard Core	HC-203	Theory of Computation	5	5	30	70	100
9	Hard Core	HC-204	a) Data Structure & Algorithm Practical b) Operating System Practical	--	5	30	70	100
10	Core Elective	CE-201	Artificial Intelligence <b>OR</b> Data Science	5	5	30	70	100
11	Open Elective	OE-201	E-Commerce <b>OR</b> MOOCs (From SWAYAM/ NPTEL etc.)	--	4	--	50	50
<b>Total</b>					<b>29</b>	<b>150</b>	<b>400</b>	<b>550</b>

Semester-III								
Sl. No.	Nature of Course	Course Code	Paper Title	Units	Credits	Marks		
						Mid-Sem.	End-Sem.	Total
12	Hard Core	HC-301	Computer Networks	5	5	30	70	100

13	Hard Core	HC-302	Software Engineering	5	5	30	70	100
14	Hard Core	HC-303	a) Computer Networks Practical b) Software Engineering Practical	--	5	30	70	100
15	Core Elective	CE-301	Compiler Design <b>OR</b> Graph Theory	5	5	30	70	100
16	Core Elective	CE-302	Cloud Computing <b>OR</b> Soft Computing	5	5	30	70	100
17	Field Internship	FI-201	Field Internship	--	3	--	50	50
	<b>Total</b>				<b>28</b>	<b>150</b>	<b>400</b>	<b>550</b>

Semester-IV								
Sl. No.	Nature of Course	Course Code	Paper Title	Units	Credits	Marks		
						Mid-Sem.	End-Sem.	Total
18	Hard Core	HC-401	Applied Cryptography	5	5	30	70	100
19	Hard Core	HC-402	a) Java Practical b) Applied Cryptography Practical	--	5	30	70	100
20	Hard Core	HC-403	Dissertation	--	5	--	100	100
21	Core Elective	CE-401	Data Mining <b>OR</b> Internet of Things	5	5	30	70	100
22	Allied Core	AC-401	Women and Society (For All PG Subjects/ Programs)	3	3	15	35	50
	<b>Total</b>				<b>23</b>	<b>105</b>	<b>345</b>	<b>450</b>

#### Summary

<b>HC-Hard Core</b>	14 x100	1400
<b>CE-Core Elective</b>	4 x100	400
<b>OE-Open Elective</b>	1x50	50
<b>AC-Allied Core</b>	2x50	100
<b>FI-Field Internship</b>	1x50	50
<b>Total Marks:</b>		<b>2000</b>

#### Summary

Semester	Credits	Total Marks
<b>Sem-I</b>	23	450
<b>Sem-II</b>	29	550
<b>Sem-III</b>	28	550
<b>Sem-IV</b>	23	450
<b>TOTAL</b>	<b>103</b>	<b>2000</b>

**DEPARTMENT OF COMPUTER SCIENCE**  
**Rama Devi Women's University, Bhubaneswar**

**M.Sc. Computer Science**

**Programme Outcomes (POs)**

After completion of the course, the student will achieve the following:

- PO1.Engineering knowledge:** Apply the knowledge of mathematics, science, and computer science specialization to evaluate, analyse, synthesize, model and integrate technologies to solve complex scientific problems.
- PO2.Problem analysis:** Analyse complex scientific problems critically, apply independent judgment for synthesizing information to make intellectual and/or creative advances for conducting research in a wider theoretical and practical context.
- PO3.Design/development of solutions:** Design and develop a system to provide a wide range of potential, feasible and optimal solutions for critical and challenging scientific problems to meet the desired needs within social areas such as economics, environmental, and ethics.
- PO4.Conduct investigations of complex problems:** Research skill to extract information pertinent to unfamiliar problems through literature survey and experiments, apply appropriate research methodologies, techniques and tools, design, conduct experiments, analyse and interpret data, demonstrate higher order skill and view things in a broader perspective, contribute individually/in group(s) to the development of scientific/technological knowledge in one or more domains of Science.
  
- PO5.Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern IT tools including prediction and modelling to complex scientific activities with an understanding of the limitations.
- PO6.The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional practice.
- PO7.Environment and sustainability:** Understand contemporary issues in providing technology solutions for sustainable development considering impact on economic, social, political, and global issues and thereby contribute to the welfare of the society.
- PO8.Ethics:** Acquire professional and intellectual integrity, professional code of conduct, ethics of research and scholarship, consideration of the impact of research outcomes on professional practices and an understanding of responsibility to contribute to the community for sustainable development of society.
- PO9.Individual and team work:** Possess knowledge and understanding of group dynamics, recognize opportunities and contribute positively to collaborative-multidisciplinary scientific research, demonstrate a capacity for self-management and team work ,decision-making based on

open-mindedness, objectivity and rational analysis in order to achieve common goals and further the learning of themselves as well as others.

**PO10.Communication:** Communicate with the science community, and with society at large, regarding complex engineering activities confidently and effectively, such as, being able to comprehend and write effective reports and design documentation by adhering to appropriate standards, make effective presentations, and give and receive clear instructions.

**PO11. Project management and finance:** Demonstrate knowledge and understanding of management principles and apply the same to one's own work, as a member and leader in a team, manage projects efficiently in respective disciplines and multidisciplinary environments after consideration of economic and financial factors.

**PO12.Life-long learning:** Recognize the need for, and have the preparation and ability to engage in life-long learning independently, with a high level of enthusiasm and commitment to improve knowledge and competence continuously.

### **Programme Specific Outcomes (PSOs)**

A graduate with a M.Sc. in Computer Science will have the ability to

**PSO1.** Communicate computer science concepts, designs, and solutions effectively and professionally.

**PSO2.** Apply knowledge of computing to produce effective designs and solutions for specific problems.

**PSO3.** Use software development tools, software systems, and modern computing platforms.

## **DETAIL SYLLABUS**

### **HC-101 DISCRETE MATHEMATICAL STRUCTURE**

#### **Course Outcomes**

- Upon successful completion of this course, students will be able to:
- **CO1.** Apply mathematical logic to solve problems and prove theorems.
- **CO2.** Understand sets, relations, functions and discrete structures.
- **CO3.** Solve counting problems by applying counting techniques, permutations, combinations, pigeonhole principle.
- **CO4.** Learn various concepts of graph theory and apply to real world problems.
- **CO5.** Understand the algebraic structure: Group, Ring, Field.

## **UNIT I**

Propositional Logic, Propositional equivalences, Rules of Inference for Propositional Logic. Predicates and Quantifiers, Nested Quantifiers, Rules of Inference for Quantified Statements. Methods of Proof: Direct Proofs, Proof by Contraposition, Proofs by Contradiction.

## **UNIT II**

Sets, Set Operations, Set Identities. Functions, One-to-One and Onto Functions, Inverse Functions and Compositions of Functions. Relations, Properties of Binary Relations, Composition of Relations, Equivalence Relations, Equivalence Classes and Partitions, Partial Orderings, Chain, Hasse Diagrams, Lattices.

## **UNIT III**

Mathematical Induction, Strong Induction. The Basics of Counting, Principle of Inclusion-Exclusion, The Pigeonhole Principle. Permutations and Combinations.

## **UNIT IV**

Introduction to Graphs, Graph Terminology, The Handshaking Theorem, Special Simple Graphs (Complete Graphs, Cycles, Wheels, Bipartite Graphs). Representing graphs, Graph Isomorphism. Euler Paths and Circuits, Hamilton Paths and Circuits, Planar Graphs, Euler's Formula for Planar Graphs.

## **UNIT V**

Groups, Subgroups, Cosets and Lagrange's Theorem. Codes and Group codes. Ring, Integral Domains and Fields.

## **Text Books**

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, McGrawHill International.
2. C.L. Liu, "elements of Discrete Mathematics", McGraw Hill International

**Mapping of Course Outcomes with the Program Outcomes:**

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	5	4	4	1	4	1	1	1	1	1	2
CO2	5	4	4	2	1	1	1	1	1	1	1	2
CO3	5	4	4	2	1	1	1	1	1	1	1	2
CO4	5	4	5	4	2	1	1	1	1	1	1	2
CO5	4	4	4	2	1	1	1	1	1	1	1	2

**HC-102      COMPUTER SYSTEM ARCHITECTURE      [30+70=100]**

**Course Outcome**

- CO 1: Understand the advanced concepts of computer architecture.
- CO 2: Analyze to the major differentials of RISC and CISC architectural characteristics.
- CO 3: Investigate modern design structures of Pipelined and Multiprocessors systems.
- CO 4: Acquainted with recent computer architectures and I/O devices, as well as the low-level language required to drive/manage these types of advanced hardware.
- CO 5: prepare selected reports that imply some emergent topics supporting material essence.

**UNIT I**

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 II**

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, Fire Wire and InfiniBand.

**UNIT III**

CPU Structure and Function: Processor Organization, Register Organization, Instruction Cycle, Instruction Pipelining.



Reduced Instruction Set Computer (RISC): Instruction Execution Characteristics, Use of a large Register File, Compiler-Based Register Optimization, Reduced Instruction Set Architecture, RISC Pipelining, MIPS R4000, SPARC, RISC Versus CISC Controversy.

#### UNIT IV

Instruction Level Parallelism and Superscalar Processors, Overview and Design Issues of Pentium-4. IA-64 Architecture: Motivation, General Organization, Prediction, Speculation, and Software Pipelining. IA-64 Instruction Set Architecture, Itanium Organization.

#### UNIT V

Parallel Organization: Multiple Processors Organizations, Symmetric Multiprocessors, Cache Coherence and MESI Protocol. Clusters, Non-Uniform Memory Access (NUMA), Vector Computation.

#### Text Books

1. Stalling W. Computer Organization and Architecture. (PHI)
2. C. Hamacher G. Vranesic, S. Zaky – Computer Organization McGraw Hill 1996

#### References

1. M. M. Mano – Computer System Architecture, 3rd Edition, PHI 1993
2. K. Hwang – Advanced Computer Architecture, McGraw Hill, 1993

CO-PO Mapping Table

POs COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO 1	5	5	4	4	4	1	1	1	1	4	1	1
CO 2	5	5	4	5	5	1	1	1	4	1	1	1
CO 3	4	5	5	5	5	1	1	1	4	4	1	1
CO 4	5	5	5	5	5	1	1	1	4	4	1	1
CO 5	5	5	5	5	5	1	1	1	5	1	4	1

## **HC-103      DATABASE SYSTEMS IMPLEMENTATION**

Course outcomes:

CO1: Analyse application data using E-R modelling and describe the logical and physical database designs.

CO2: Understand relational algebra, calculus and apply structured query language (SQL) for database definition and manipulation.

CO3: Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database.

CO4: Use transaction management systems

CO5: Use Concurrency Control methods and apply security on database systems.

### **UNIT I:**

Introduction to Database, 3-Schema Architecture and Data independence, 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, Refining the ER design for the company database, ER diagram Naming conventions and design issues.

ER to relational mapping and Enhanced Entity-Relationship (EER) and object modeling subclasses.

### **UNIT II:**

Relational Algebra and Calculus: Relational Algebra operations, Tuple relational calculus, Domain relational calculus. SQL- The Relational Database Standard: Data Definition, Constraints and schemas, Insert, Delete and Update statement in SQL. SQL queries: Basic and complex SQL queries, Aggregate functions.

### **UNIT III:**

Database Design Theory: Functional Dependencies, Armstrong's Axioms. Closure of attributes, Dependency preservation, Lossless design. Normalization: Normal Forms on Primary Keys, Second and third Normal Forms, Boyce-Codd Normal Form.

### **UNIT IV:**

Query processing and Optimization: Translating SQL queries into relational algebra, Basic

diagram for executing query operations, Using Heuristics in query optimization. Transaction processing concepts: Introduction, Transaction and system concepts, Desirable properties of transaction.

Schedules and recoverability: Types of Schedule, Serializability of schedules, Checking serializability of schedules.

**UNITV:**

Concurrency Control Techniques: Locking techniques for concurrency control, Concurrency control based on time stamp ordering, Multi version concurrency control techniques, Validations concurrency control techniques. Database Security and Authorization: Introduction to database security issues, Discretionary access control based on granting and revoking privileges, Mandatory access control for multilevel security.

Distributed Database Systems: Client Server architecture, Distributed database concepts, Data fragmentation, Replication and allocation technique for distributed database, Types of distributed database systems.

**Text Books**

1. RamezElmasri and Shamkant B. Navathe, “*Fundamentals of Database Systems*”, Pearson Education, 7<sup>th</sup> Ed.,2016.
2. Rajeeb C. Chatterjee, “*Learning Oracle SQL and PL/SQL: A simplified Guide*”, PHI Learning Private Limited,2012.

**Reference Books**

1. A.Silberschatz,H.F.Korth,S.Sudarshan,“*DatabaseSystemConcepts*”,McGrawHill,7<sup>th</sup>Ed.,2021.
2. Raghu Ramakrishnan and Johannes Gehrke, “*Database Management Systems*”, McGraw Hill, 3<sup>rd</sup> Ed., 2014.

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	5	4	4	1	1	1	1	1	1	1	1
CO2	5	1	4	1	1	1	1	1	1	1	1	1
CO3	4	1	4	1	1	1	1	1	1	1	1	4
CO4	4	1	1	1	4	1	1	1	2	2	4	4
CO5	4	1	1	1	4	1	1	1	2	2	2	4

**(a) Database Systems Practical**

**(b) Python Practical**

**Course Outcomes**

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

CO 1: Ability to understand the various kinds of SQL commands

CO 2: Demonstrate the operation on database table.

CO 3: Ability to make customized query efficiently on a database.

CO 4: Ability to apply query techniques for realistic data.

CO 4: Write, test, and debug simple Python programs.

CO 5: Ability to understand Python code, develop medium difficulty applications in Python

**List of Experiments**

**a) Database System Practical**

1. Creating/Altering/Deleting Tables.
2. Retrieving data using SQL Queries.
3. Creating tables with constraints.
4. Working on Multiple tables with join.
5. WAP a PL/SQL program to find the factorial of a number.
6. WAP a PL/SQL program to print the Fibonacci series upto n terms.

**b) Python Practical**

1. Write a python program to convert temperature from Celsius to Fahrenheit and vice versa
2. Write a python program to find largest of three numbers.
3. Write a program to compute distance between two points taking input from the user (Pythagorean Theorem).
4. Write a python program to print prime numbers less than 100.
5. Write a python program to find factorial of a number using recursion.
6. Create a list and perform the following methods
  - a) insert()
  - b) remove()
  - c) append()
  - d) len()
  - e) pop()
  - f) clear()

PO-CO Mapping Table

POs COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO 1	5	5	5	2	2	1	1	1	1	1	1	5
CO 2	5	5	5	2	2	1	1	1	1	1	1	5
CO 3	5	5	5	2	2	1	1	1	1	1	1	5
CO 4	5	5	5	2	2	1	1	1	1	1	1	5
CO 5	5	5	5	2	2	1	1	1	1	1	1	5

**HC-201 SEMESTER-II DATA STRUCTURE & ALGORITHM**

## **Course Outcomes**

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

**CO1.** Analyze the efficiency of algorithms using asymptotic notations, apply divide-and-conquer method to design algorithms, and solve recurrences.

**CO2.** Use standard data structures like hash tables, heaps, and trees to store data efficiently.

**CO3.** Explain major graph algorithms and apply these algorithms to solve real world problems.

**CO4.** Design and analyze algorithms using greedy technique and dynamic programming.

**CO5.** Understand complexity classes P, NP, co-NP, NP-hard, NP complete and NP complete reductions.

### **UNIT I**

Introduction to algorithms, Analysis of Insertion sort. Asymptotic Notations, Divide and Conquer Approach, Merge Sort, Recurrence Relations. Solving Recurrences: Substitution methods, Recursion tree method, and Master method.

### **UNIT II**

Quick Sort, Heap Sort. Hash Tables, Hash Functions, Chaining, Open Addressing: Linear probing, Quadratic probing and Double hashing Binary search trees, AVL tree, B-trees.

### **UNIT III**

Representation of Graphs, Breadth-First Search, Depth-First search, Topological sort, Minimum spanning trees: Prim's and Kruskal's Algorithm. Single source shortest paths: The Bellman-Ford algorithm, Dijkstra's algorithm.

### **UNIT IV**

Dynamic programming: Matrix Chain multiplication, Longest Common Subsequence.

Greedy algorithms: Activity selection problem, Human codes.

Amortized analysis: Aggregate Analysis, The Accounting Method, The Potential Method.

### **UNIT V**

Decision Problems vs. Optimization Problems, Polynomial time, Polynomial-time Verification, Polynomial-time Reduction

Complexity Classes: P, NP, co-NP, NP-hard and NP-complete. Example of NP-Complete Problems (Satisfiability problem, 3-CNF satisfiability problem, Clique problems, Vertex cover problem, Hamiltonian cycle problem, Travelling salesman problem).

NP-Completeness proofs: Clique to vertex cover problem reduction, Hamiltonian cycle to Travelling salesman problem reduction.

## **Text Books**

1. Introduction to Algorithms, T.H. Corman, C.E.Leiserson, R.L.Rivest and C.Stein
2. Algorithm Design, Jon Kleinberg, ÉvaTardos.

**Mapping of Course Outcomes with the Program Outcomes:**

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	5	4	4		1	1	1	1	1	1	2
CO2	4	4	4	4	2	1	1	1	1	1	1	2
CO3	5	4	5	5	2	1	1	1	1	1	1	2
CO4	5	5	5	5		1	1	1	1	1	1	2
CO5	5	5	4	5	2	1	1	1	1	1	1	2

**HC-202 OPERATING SYSTEMS**

**Course Outcomes**

**CO1:** Analyze the concepts of Operating System.

**CO2:** Analyze the concepts of process, thread and deadlock situation and Illustrate the Scheduling of a processor for a given problem instance.

**CO3:** Analyze memory management techniques and implement page replacement Algorithm.

**CO4:** Understand the implementation of file systems and directories.

**CO5:** Understand the implementation of files system using UNIX operating system.

**UNIT-I**

Introduction: Definition of OS, History of DOS and UNIX Operating

System Process: Definition of Process, Process States.

Deadlock: Definition of Deadlock, Causes of Deadlock, Avoidance of Deadlock, and Recovery from Deadlock.

**UNIT-II**

Processor management: Preemptive versus non-preemptive scheduling - priorities -deadline scheduling - FIFO - RR -SJF

Distributed computing: Classification of sequential and parallel processing-array processors. Dataflow computers - multiprocessing - fault tolerance.

**UNIT-III**

Memory Management: Background, Logical versus Physical Address space, swapping, contiguous Allocation. Paging, Segmentation.

Virtual Memory: Background, Demand paging, performance of Demand paging.

Page Replacement: Page Replacement Algorithms. Allocation of frames, Thrashing, Demand Segmentation.

#### UNIT-IV

Device and information management: Operation of moving head disk storage - need for disk scheduling. Optimization - FCFS - SSTF - SCAN - RAM disks - optical disks.

Files and database systems: File system - function - organization - allocating and freeing space - file descriptor - access control matrix.

#### UNIT-V

Case studies: DOS - memory management - overlaying - extended and expanded memory - memory allocation.

File system and allocation method - internal and external command memory management functions - file management functions.

UNIX: Process in UNIX - memory management - I/O systems - file systems and allocation method.

#### Text Book

1. Willam Stallings, "Operating Systems", 5/e PHI/Pearson Education.
2. Silberschatz, Peterson, Galvin, "Operating System Concepts", Addison Wessely, Fifth Edition,
3. H. M. Deital, "An introduction to operating systems", Addison Wisely,

#### References

1. Charles Crowley, "Operating systems - A Design Oriented Approach", Tata McGrawHill.
2. Andrew S. Tannenbaum, "Operating Systems: Design and Implementation", PHI,2006.
3. Mukeshsinghal, Niranjan G shivaratri, "Advanced concepts in operating systems",MGH.

#### Mapping of Course Outcomes with the Program Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	2	5	4	4	2	1	1	1	1	1	1
CO2	5	4	4	4	4	2	1	1	1	1	1	1
CO3	4	4	5	5	2	2	1	1	1	1	1	1
CO4	4	5	5	5	2	2	1	1	1	1	1	1
CO5	2	5	4	5	2	2	1	1	1	1	1	1



## HC-203 THEORY OF COMPUTATION

### COURSE OUTCOMES:

After completion of the course, a student will be able to

- CO 1 Describe the concept of abstract machines and their power to recognize the languages.
- CO 2 Apply finite state machines for modelling and solving computing problems.
- CO 3 Design context free grammars for formal languages.
- CO 4 Distinguish between decidability and undesirability.
- CO 5 Solve mathematical tools and formal methods
- CO 6 Apply mathematical and formal techniques for solving problems in computer science

### **UNIT-I:**

Background materials: Alphabets, Strings, Empty Strings, Sets, Empty Set,

Proof Methods: Induction, Contradiction, Hypothesis

Introduction to Theory of Computation: Finite State Machine, Deterministic Finite Automata, Non-deterministic Finite Automata, Equivalence of NFA and DFA, Minimization of Finite state Machine

### **UNIT-II:**

Regular Expressions and Languages: Introduction, Conversion of DFAs to Regular Expressions, and Vice versa, Pumping Lemma

Closure Properties of Regular Languages: Union, Intersection, Complement, Difference, Reversal, Homomorphism, and Inverse Homomorphism.

Context Free Languages: Context Free Languages, Context Free Grammars, Derivation, Ambiguity,

### **UNIT-III:**

Push Down Automata: Definition of PDAs, Acceptance of PDAs by final state and by empty stack, Conversion of CFG to PDA and vice versa, DPDA and NPDA

Simplification of CFG: Chomsky Normal Form, Greibach Normal Form. The Pumping Lemma for CFL's. Closure properties CFL.

Turing Machines: TM Definition and Notation, Instantaneous Descriptions, NTM & DTM,

Extensions and Restrictions to Basic TM Model: Multi Tape, Multi Dimensional, Counter machine, Two StackPDAs.

### **UNIT-IV:**

Decidability Theory: The Church-Turing Thesis, Universal Turing Machines and TM Encoding, Decidable and Semi-decidable languages: Recursive Enumeration and Decidability, Many-one Reductions, Hardness, Undecidability

Language Properties: Closure Properties, The Diagonalization Language, The Halting Problem, Post's Correspondence Problem, Undecidable Problems from Language Theory, Linear Bounded /automata (LBA).

**UNIT-V:**

Complexity Theory: Measuring Complexity, The O, Ω, Θ notations Time

Complexity classes: P, NP, NP – Completeness,

Some NP-Complete Problems: SAT, 3-SAT, Hamiltonian Path, Vertex Cover,

Independent Set, Space Complexity classes: PSPACE, L, NL.

**Text Book**

1. Introduction to Automata Theory, Languages & Computation – Hopcroft, Motwani & Ullman.
2. Introduction to Theory of Computation: M.Sipser, Thomson Learning

**Reference Book**

1. Fundamental of the theory of Computation ,Principles and Practice- R.Greenland,H.J.Hoorer.
2. K.L.P. Mishra and N. Chandrasekaran, Theory of Computer Science, PHI.
3. Michael Sipser, Introduction to the Theory of Computation, Thomson Learning.

CO-PO-PSOMAPPING

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO2	PSO 3	PSO 4
AEC3.1	5	1	1	1	1	1	1	1	1	1	1	1	1	4	1	4
AEC3.2	1	5	1	4	1	1	1	1	1	1	1	1	4	5	4	1
AEC3.3	1	1	4	5	1	1	1	1	1	1	1	1	4	5	4	5
AEC3.4	2	4	4	1	5	1	1	1	1	1	1	1	1	4	4	5
AEC3.5	2	1	1	1	5	1	1	1	1	1	1	1	5	1	5	1
AEC3.6	1	5	1	4	1	1	1	1	1	1	1	1	1	1	1	4
<b>AEC3</b>	2	4		4	1	1	1	1	1	1	1	1	4	4	4	4

(a) **Data Structure &Algorithm Practical**

(b) **Operating System Practical**

**Course Outcomes**

Upon Completing the Course, Students will able to

- CO 1: For a given sorting problem (Merge/Insertion/Quick) student will able to implement it and analyze the same to determine the time and computation Complexity.
- CO 2:Student will able to implement program for Graph traversal Algorithm & Hashing Technique.
- CO 3:Implements various OS Scheduling Algorithms.
- CO 4: Implements various Memory Scheduling Algorithms
- CO 5:Emphasize hands-on experience working with various algorithm associated with Data Structure and Operating Systems

**List of Experiments**

a) **Data Structure &Algorithm**

1. WAP in C to implement Quick Sort
2. WAP in C to implement Insertion Sort
3. WAP in C to implement Merge Sort
4. WAP in C to implement Binary Search Tree
5. WAP in C to implement Shortest Path Method (Dijkstra's Method)
6. WAP in C to implement Implementing Hashing (Linear &Quadratic)

b) **Operating System Practical**

1. Implementing FCFC Scheduling.
2. Implementing SJFS scheduling.
3. Implementing Priority Scheduling.
4. To Simulate MFT Memory Management Technique
5. To Simulate Memory Management Technique
6. To Simulate FIFO page replacement

PO-CO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	4	4	1	1	1	1	1	1	1	5
CO2	2	4	4	4	5	1	1	1	1	1	1	5
CO3	2	4	2	2	5	1	1	1	1	1	1	5
CO4	2	2	2	2	1	1	1	1	1	1	1	5

## **CE-201 ARTIFICIAL INTELLIGENCE**

### **COURSE OUTCOMES:**

After completion of the course, a student will be able to

- CO 1 Study the concepts of AI and related searching algorithms.
- CO 2 Develop the knowledge skills and its representational structure in AI
- CO 3 Study the concepts of natural language processing in AI.
- CO 4 Study the concepts of supervised/unsupervised machine learning and game technique.
- CO 5 Study how design the programming skill in PROLOG, and concepts of pattern recognition approaches.

### **UNIT I**

Introduction to AI: Foundations of AI, History of AI, State of Art Intelligent agents: Agents and Environments, The concept of Rationality, Structure of Intelligent Agent Problem Solving by Searching: Problem-Solving Agents, Searching for Solutions, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Heuristic Functions

### **UNIT II**

Beyond Classical Search: Local Search Algorithms and Optimization Problems, Searching with Nondeterministic Actions, Searching with Partial Observations. Adversarial Search: Games, Optimal Decisions in Games, Alpha-Beta Pruning Constraint Satisfaction Problems: Defining Constraint Satisfaction Problems, Local Search for CSPs, The Structure of Problems.

### **UNIT III**

Logical Agents: Knowledge-Based Agents, The Wumpus World, Logic and Propositional Logic First-Order Logic: Syntax and Semantics of First-Order Logic, First-Order Logic, Knowledge Engineering in First-Order Logic Inference in First-Order Logic: Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining and Backward Chaining, Resolution

### **UNIT IV**

Classical Planning: Definition of Classical Planning, Planning Graphs, Other Classical Planning Approaches Knowledge Representation: Ontological Engineering, Categories and Objects, Events, Mental Events and Mental Objects, Reasoning Systems for Categories Quantifying Uncertainty: Acting under Uncertainty, Basic Probability Notation, Bayes' Rule and its use

### **UNIT V**

**Learning from Examples: Forms of Learning, Supervised Learning, Learning Decision Trees  
 Reinforcement Learning: Introduction, Passive Reinforcement Learning, Active Reinforcement  
 Learning, Generalization in Reinforcement Learning Natural Language Processing: Language Models,  
 Text Classification, Information Retrieval, Information Extraction**

**Text Book**

1. Stuart Russel & Peter Norvig: Artificial Intelligence A Modern Approach. (Person Education Asia.)
2. Artificial Intelligence – Mishra, PHI
3. D.W. Patterson, “Introduction to A.I and Expert Systems”, PHI,
4. Rich & Knight, “Artificial Intelligence”, Tata McGraw Hill,.
5. W.F. Clocksin and Mellish, “Programming in PROLOG”, Narosa Publishing House, 3/e

**CO-PO MAPPING**

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	4	5	5	4	4	2	2	1	1	1	1	
CO 2	2	4	5	2	4		2	1	1	1	1	1
CO 3	2	2		2	2		5	1	1	1	1	1
CO 4	4	2	4	5	2		2	1	1	1	1	1
CO 5	2	4	5	5	4		2	1	1	1	1	1

**CE-201 DATA SCIENCE**

**Course Outcomes**

CO 1: To explicate data analysis techniques and quantitative modelling for the solution of real-world

Business problems.

CO 2: To report findings of analysis and effectively present using data visualization techniques.

CO 3: To demonstrate knowledge of statistical data analysis techniques utilized in business decision making.

CO 4: To provide insights about the roles of a Data Scientist, such as a developer, an analyst, a statistical

Expert etc.

CO 5: To understand the techniques and tools for transformation of data

## **UNIT I**

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

## **UNIT II**

Overview of R, R data types and objects, reading and writing data.  
Control structures, functions, scoping 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 from colleagues in various  
formats. Basics of data cleaning and making data "tidy".

## **UNIT IV**

Exploratory Data Analysis, Essential exploratory techniques for summarizing data, applied for  
formal modelling 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.

## **UNIT V**

Make beautiful visualizations using the ggplot2 library  
Create commonly used data visualizations for each data type including histograms, scatter plots,  
and box plots, improve your data visualizations using facets.  
Create reference variables using appropriate scope, Use the popular diamonds dataset to put your R  
skills to work.

### **Text Books**

1. Rachel Schutt, Cathy O'Neil, "Doing Data Science: Straight Talk from the Frontline" by Schroff/O'Reilly,2013.
2. Foster Provost, Tom Fawcett, "Data Science for Business" What You Need to Know About Data Mining and Data-Analytic Thinking by O'Reilly,2013.

### **Reference Books**

1. 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", 1<sup>st</sup> Edition, by Wiley,2013.

PO-CO MAPPING TABLE

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	4	4	1	1	1	1	1	1	1	2
CO2	2	4	4	4	1	1	1	1	1	1	1	2
CO3	2	4	2	2	1	1	1	1	1	1	1	2
CO4	2	4	2	2	1	1	1	1	1	1	1	2
CO 5	2	4	2	2	1	1	1	1	1	1	1	2

## OE-01 E-COMMERCE

### Course Outcomes

CO:1 To explicate data analysis techniques and quantitative modelling for the solution of real-world

Business problems.

CO 2: To report findings of an analysis and effectively present them using data visualization techniques.

CO 3: To demonstrate knowledge of statistical data analysis techniques utilized in business decision making.

CO 4: Applying the e-commerce concept in business world

### UNIT I

Introduction: -Overview of Electronic Commerce - Definition of Electronic Commerce - E Business- Categories of E-Commerce Application Benefits: - Potential benefits of E Commerce - Advantages and Disadvantages of E-commerce- Impact of E Commerce on Business –Global trading Environment of E-commerce- The global information distribution networks Legal and Ethical Issues: - The regulatory environment for E Commerce-Issues related to E-commerce- Legal Issues

### UNIT II

Business Model of E-commerce, Internet Trading Relationships- Consumer to Business (C2B)

Business to Consumer(B2C)

Business to Business (B2B), Consumer to Consumer (C2C)-Business to Government

(B2G) Difference between B2C and B2B E-Commerce, Advantages and

Disadvantages.

### UNIT III

E-Commerce and EDI, introduction to Electronic Data Interchange (EDI)- Benefits Features of EDI – EDI Model- EDI Standards - Data Transfer and Standards – Cost of EDI Electronic Funds Transfer–EFT-Combining EDI and EFT- Roles of Banking EDI

### UNIT IV

Security Overview: Cryptography and Authentication Introduction - Messaging Security Issues Confidentiality - Integrity - Authentication. Encryption Techniques, Digital Signatures Good Encryption Practices – Key Management - key management tasks – Additional Authentication Methods. Firewalls -Definition - component - Functionality - securing the firewall - factors considered in securing the firewall -Limitations.

#### Text Books

1. Electronic Commerce - Security, Risk Management and Control, Greenstein and Feinman.
2. Electronic Commerce: From Vision to Fulfillment, Elias M Award(PHI)

#### Reference Books

1. E-Commerce and Mobile Commerce Technology- U.S. Pandey, Saurabh Shukla, S.Chand
2. Electronic Commerce - A Managerial Perspective, Ed-aim Turban, Jae Lcc, David King Michael Chung, Addison Wesley, NewDelhi.

PO-CO MAPPING TABLE

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	4	4	1	1	1	1	1	1	1	2
CO2	2	4	4	4	5	1	1	1	1	1	1	2
CO3	2	4	2	5	5	1	1	1	1	1	1	2
CO4	2	2	2	5	1	1	1	1	1	1	1	2

**SEM-III HC-301 COMPUTER NETWORKS**

#### Course Outcomes

CO1: Identify data communications system components, network topologies, and protocols.



CO2: Analyze different features of analog and digital transmission.

CO3: Analyze the working principles and protocols of data link layer.

CO4: Identify and differentiate working principles and protocols of network and transport layer.

CO5: Identify and implement different types of application in application layer.

## **UNIT I**

Overview of Data Communications: Network Topologies, Reference Models: OSI Model. Physical Layer: Analog and Digital Signals, Data Rate Limits, Transmission Impairment. Digital Transmission: Line Coding, Sampling, Transmission Modes.

## **UNIT II**

Analog Transmission: Modulation of Digital Data, Modulation of Analog signals.

Multiplexing: Frequency Division Multiplexing (FDM), Wave Division Multiplexing (WDM), Time Division Multiplexing(TDM). Switching Techniques: Circuit Switching and Packet Switching.

## **UNIT III**

Errors: Types of Errors, Error Detection, Error Correction. Data Link Control and Protocols: Stop-and-Wait ARQ, Go-Back-N ARQ, Selective Repeat ARQ Wireless LANs: IEEE 802.11 and its architecture

## **UNIT IV**

Host to Host Delivery: IP Addressing and Routing: Unicast, Multicast, Broadcast, and Anycast

Network Layer Protocols: Address Resolution Protocol (ARP), IPV4, Internet Control Message Protocol (ICMP), IPV6. Transport Layer: User Datagram Protocol (UDP), Transmission Control Protocol (TCP)

## **UNIT V**

Client Server Model: Domain Name System (DNS): Electronic Mail (SMTP).

File Transfer: File Transfer Protocol (FTP), Post Office Protocol (POP), HyperText Transfer Protocol (HTTP) and World Wide Web(WWW) Network Security: Authentication, Digital Signatures and Certificates, Firewalls.

### **Text Books**

1. Data Communications and Networking: Behrouz A. Forouzan, Tata McGraw-Hill, 4<sup>th</sup>Edition.
2. Computer Networks: A. S. Tannenbum, D. Wetherall, Prentice Hall,5<sup>th</sup>Edition.

### **Reference Books**

1. Data and Computer Communications: William Stallings, Prentice Hall, 9<sup>th</sup> Edition
2. Data Communication and Computer Networks: Ajit Pal, PHI Learning Pvt.Ltd

PO-CO MAPPING TABLE

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	2	2	4	4	1	1	1	1	1	1	2
CO2	4	4	4	4	5	1	1	1	1	1	1	5
CO3	2	4	2	2	5	1	1	1	1	1	1	5
CO4	2	2	2	2	2	1	1	1	1	1	1	5
CO5	2	4	4	4	4	1	1	1	1	1	1	1

## HC-302 SOFTWARE ENGINEERING

### Course Outcomes:

CO 1: To provide the idea of decomposing the given problem into Analysis, Design, Implementation,

Testing and Maintenance phases.

CO 2: To provide an idea of using various process models in the software industry according to given circumstances.

CO 3: To gain the knowledge of how to gather and specify requirements of software projects.

CO 4: To differentiate different testing methodologies and their utilities.

CO 5: To understand and apply the basic project management practices in real life projects

CO 6: To enhance the ability to work in a team as well as independently on software projects

### UNIT I

Computer-Based system Engineering: Emergent System Properties, Systems and their Environment, System Modeling, System Engineering Process, System Procurement. Software Process: Software Process Models, Process Iteration, Software Specification, Design and Implementation. Software Validation and Evaluation, Automated Process Support.

### UNIT II

Software Requirements: Functional and Non-Functional Requirements, Use Requirements. System Requirements, Software Requirements Document. Requirements Engineering Processes: Feasibility Studies, Requirements Elicitation and Analysis, Requirements Validation, Requirements Management. System Models: Context Models, Behavioral Models, Data models, Object models.

### UNIT III

Architectural Design: System Structuring, Control Models, Modular Decomposition, Domain-specific Architectures. Distributed System Architectures: Multiprocessor

Architectures, Client-Server Architectures, Distributed Architectures, CORBA. Patterns. Dependability: Critical Systems, Availability and Reliability, Safety, Security, Critical Systems

#### UNIT IV

Specifications: Software Reliability Specification, Safety Specification, Security Specification. Critical Systems Development: Fault Minimization, Fault tolerance, Fault Tolerance Architectures, Safe System Design. Object Oriented software design, Development process.

#### UNIT V

UML: Unified Modeling Language, Use case diagram, Class diagrams, essential Interaction diagram, Object diagram, Packages and collaboration. State diagrams, Activity diagrams, Physical diagrams.

#### Text Books

1. Software Engineering – Sommerville, Addition-Wesley
2. Software Engineering- Pressman, R.S, MGH

#### Reference Books

1. Fundamental of Software Engineering—Rajib Mall, PHI
2. Software Engineering- Agarwal, K.K & Singh, New Age International.

PO-CO MAPPING TABLE

POs COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1.	4	4	1	1	4	4	1	1	4	5	4	4
CO 2	4	5	4	4	5	1	1	1	1	4	1	1
CO 3	4	5	5	5	5	1	1	1	4	4	1	1
CO 4	5	5	4	5	5	1	1	1	4	4	1	5
CO 5	1	4	4	1	4	1	1	1	5	4	4	1

- a) Computer Networks Practical
- b) Software Engineering Practical

**Course Outcome**

CO 1: Acquire knowledge of Networking Parameters

CO 2: Able to do establish Client/Server Communication using Socket

CO 3: Learn how to use File Transfer protocol

CO 4: Learn to use Software Engineering Tools to develop various automated systems.

CO5: Using UML tools to analyse project work

**a) Computer Networks****List of Experiment (Using C/C++/Java)**

1. Listing Network Interface Properties
2. Getting the ownIP.
3. Implementing Ping Program.
4. Implementation of port scanner
5. Implements TCP Sockets for displaying date & time from server
6. Implementing File Transfer.

**b) Software Engineering (Using sample case study on Bank ATM System)**

1. Introduction To software engineering
  - a. Defining Problem definition
  - b. Writing SRS
    - i. Introduction
    - ii. Overall Description
    - iii. Specification Requirement
    - iv. Front End ,Back End Specification
    - v. Data Structure
    - vi. DFD
    - vii. Testing
    - viii. Sample screen
2. OO Analysis & Design Using UML (Sample diagrams)
  - a. UseCase
  - b. Class diagram

- c. State chart diagram
- d. Activity Diagram
- e. Collaboration Diagram
- f. Package Diagram

PO-CO MAPPING TABLE

POs COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO 1	2	4	1	1	1	1	1	1	1	1	1	2
CO 2:	2	4	4	4	1	1	1	1	1	1	1	2
CO 3	2	2	2	2	2	1	1	1	1	1	1	2
CO 4	2	2	2	2	2	1	1	1	1	1	1	2
CO 5	2	4	2	4	2	1	1	1	1	1	1	2

### **SEM-III CEC-301 COMPILER DESIGN**

#### **COURSE OUTCOMES:**

After completion of the course, a student will be able to

- CO 1 Understand the internal steps of compiler.
- CO 2 Understand the fundamental concepts of formal language.
- CO 3 Implementation of top down and bottom up parsers.
- CO 4 Understand the usage of lex and yacc tools.
- CO 5 Understand SDD, SDT, intermediate code generation and machine code generation.

#### **UNIT I:**

Introduction to Compiler Design: Introduction to Compiler Design, Phases of Compiler Design, Compiler Construction Tools

Lexical Analysis: DFA, NFA, Regular Expression, Equivalent to NFAs, Minimizing the States of DFA CFG: Basics of CFG, Normal Forms, Implementation of Lexical Analyser.

#### **UNIT II**

Syntax analysis: Top down parsing concepts-Recursive Descent Parsing, FIRST and FOLLOW, LL(1) Grammars Left Recursion Elimination, Top-Down Recursive-Descent parsing

Bottom-Up Parsing: Reduction, Finding Handles, Shift-Reduce Parsing, Conflicts during Shift-Reduce Parsing

LR Parsers: Items and the LR(0) Automaton, The LR-Parsing Algorithms, SLR, CLR, LALR, Ambiguous Grammars

#### **UNIT III**

Syntax-Directed Translation: Syntax Directed Definitions, Evaluating Orders for SDD, Applications of Syntax –Directed Translation Scheme  
 Intermediate code generation: Variations of Syntax, Three-Address Code- Addresses and Instructions, Quadraples, Triples  
 Translation of Expression: Operation within Expressions, Incremental Translation, Addressing Array Elements, Translation of Array References

**UNIT IV**

Type Checking: Rules for Type Checking, Type Conversions, Overloading of Functions and Operators Control Flow: Boolean Expressions, Short-Circuit Code, Flow of Control Statements, Control Flow Translation of Boolean Expressions, Backpatching  
 Run Time Environment: Storage organization, Stack Allocation of Space, Heap Management

**UNIT V**

Code Generation: Issues in the Design of Code Generation, Addresses in the Target Code, Basic Blocks and Flow Graphs  
 Code optimization: Optimization of Basic Blocks, Peephole Optimization, Register Allocation and Assignment.  
 Machine Independent Optimization: The Principle of Source of Optimization, Introduction to Dats Flow Analysis, Introduction to Inter procedural Analysis

**Text Books**

1. AlfredAho,RaviSethi,JeffyD.Ullman,“Compilers-Principles,TechniquesandTools”,Pearson
2. ChattopadhyaySanthanu, “Compiler Design”,PHI.
3. HolubAllen, “Compilers in C”,PHI

CO-PO-MAPPING

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	4	1	1	4	1	2	1	1	1	1	1	4
CO 2	4	1	4	1	4	1	1	1	1	1	1	4
CO 3	4	4	1	4	1	4	1	1	1	1	1	4
CO 4	4	1	4	1	4	1	1	1	1	1	1	4
CO 5	2	1	1	4	1	5	1	1	1	1	1	4

**Course Outcomes**

- CO1. Explain the concept of formal graph-theoretic definitions, notations, apply Handshaking theorem and Havel-Hakimi theorem, learn about graph isomorphism.
- CO2. Find shortest paths in graphs, understand connectedness in graphs, and define Eulerian graphs, Hamiltonian graphs.
- CO3. Learn about trees and tree traversal algorithms, apply algorithms to find minimum spanning trees.
- CO4. Understand graph planarity, find geometric and combinatorial dual, learn about matching and coverings in graphs,
- CO5. Define Independent set, and clique in a graph and understand graph colouring in detail

**UNIT I**

Graph Terminologies, Simple Graph, Multigraph, Special Graphs, Complement Graph, Regular Graph, Bipartite Graphs, Subgraphs: Proper Subgraph, Spanning Subgraph, Induced Subgraph, Vertex-degrees, Handshaking Theorem, Graphic Sequences, Havel-Hakimi Theorem. Graph Representation: Adjacency Matrix, Incidence Matrix, Adjacency List, Graph Isomorphism.

**UNIT II**

Connected Graphs, Disconnected Graphs and Components, Cut-vertices, Cut-edge, Blocks, Cut-set. Weighted Graphs, Shortest Paths, Dijkstra's Algorithm. Eulerian Graphs, Hamiltonian Graphs.

**UNIT III**

Trees, Properties of Trees, Pendant Vertices in a Tree, Distance and Centers in a tree, Rooted and Binary Trees. Counting Tree, Spanning Tree, Minimum Spanning Tree, Prim's Algorithm, Kruskal's Algorithm. Tree Traversal: Pre-order, Post-order and In-order Traversal.

**UNIT IV**

Combinatorial Vs Geometric Graphs, Planar Graphs, Kuratowski Graphs, Detection of Planarity. Geometric and Combinatorial Dual, Thickness and Crossings. Matchings, Matchings and Coverings in Bipartite Graphs, Perfect Matching.

**UNIT V**

Independent Set, Clique. Graph Coloring, Chromatic Number, Chromatic Partitioning, Greedy

Coloring Algorithm, Coloring of Chordal Graphs, Brooks Theorem. The Four Colour Conjecture and Five-Colour Theorem.

**Text Books**

1. Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science.
2. D.B. West, Introduction to Graph Theory.
3. J. A. Bondy and U. S. R. Murty: Graph Theory.
4. Jon Kleinberg and Eva Tardos, Algorithm Design.

**Reference Books**

1. T.H. Corman, C.E.Leiserson, R.L.Rivest and C. Stein, Introduction to Algorithms.

PO-CO MAPPING TABLE

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	4	2	4	2	1	1	1	1	1	1	2
CO2	5	4	4	5	2	1	1	1	1	1	1	2
CO3	5	4	4	5	2	1	1	1	1	1	1	2
CO4	5	4	4	5	2	1	1	1	1	1	1	2
CO5	5	4	4	5	2	1	1	1	1	1	1	2

**SEM –III                    CEC-302                    CLOUD COMPUTING**

**Course Outcomes:**

- CO1:** Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges of Cloud.
- CO2:** Develop applications using various models and services in cloud computing.
- CO3:** Understand virtualization and outline their role in enabling the cloud computing system model. and implement different load balancing algorithms in cloud.
- CO4:** Explain Service Management in Cloud Computing.
- CO5:** Understand security mechanisms implemented at different levels.

**UNIT I:**

Overview of Computing Paradigm: Recent trends in Computing: Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing.

Introduction to Cloud Computing: Introduction, History of Cloud Computing, Characteristics of cloud computing, Benefits and limitations of Cloud Computing.

Cloud Service Providers (CSPs), Cloud Data Centres, Components of data Centres, Cloud Computing applications.



## **UNIT II:**

Cloud Computing Architecture: Comparison with traditional computing architecture (client/server), Services provided at various levels.

Service Models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), How Cloud Computing Works.

Deployment Models: Public cloud, Private cloud, Hybrid cloud, Community cloud, Case study of NIST architecture.

## **UNIT III:**

Virtualization: Introduction, Characteristics of Virtualized Environments.

Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization.

Load balancing in Cloud Computing: Importance of load balancing, Types of load balancing, Load balancing algorithms.

## **UNIT IV:**

Case Studies: Case Study of Service Model using Google App Engine, Microsoft Azure, Amazon EC2.

Service Management in Cloud Computing: Service Level Agreements (SLAs), Billing & Accounting.

Comparing Scaling Hardware: Traditional vs. Cloud, Types of Scaling, Economics of Scaling.

## **UNIT V:**

Cloud Security: Infrastructure Security: Network level security, Host level security, Application level security.

Data security and Storage: Aspects of data security, Provider data and its security, Data security issues. Jurisdictional issues raised by data location, Authentication in Cloud, Methods of Authentication.

### **Text Books**

1. Rajkumar Buyya, James Broberg, Andrzej, M. Goscinski, “*Cloud Computing Principles & Paradigms*”, Wiley,2013.
2. Tim Mather, SubraKumaraswamy, ShahedLatif, “*Cloud Security and Privacy*”, O’ Reilly, First Edition,2011.

### **Reference Books**

1. RajkumarBuyya, Christian Vecchiola, S. ThamaraiSelvi, “*Mastering Cloud Computing*”, McGraw Hill Education,2018.
2. Barrie Sosinsky, “*Cloud Computing Bible*”, Wiley,2011.
3. Nick Antonopoulos, Lee Gillam, “*Cloud Computing: Principles, Systems and Applications*”, Springer,2010.

PO-CO MAPPING TABLE

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	5	4	4	2	4	1	1	1	2	1	1	4
CO2	4	5	4	4	2	1	1	1	4	1	4	2
CO3	2	1	1	1	1	1		1	1	1	4	4
CO4	4	1	1	1	1	1	1	1	1	1	1	2
CO5	4	1	1	1	1	1	1	1	4	1	4	4

**SEM –III                      CEC-302                      SOFT COMPUTING**

**COURSE OUTCOMES**

**CO1:** Describe human intelligence and AI and explain how intelligent system works.

**CO2:** Apply basics of Fuzzy logic and use of heuristics based on human experience

**CO3:** Describe with genetic algorithms and other random search procedures useful while seeking global optimum in self-learning situations.

**CO4:** Identify the issues in multi-objective optimization problems and apply different multi-objective optimization techniques.

**CO5:** Apply different ANN techniques to real world problems

**UNIT I:**

Introduction: Concept of computing systems. Hard computing, Soft computing, Hybrid computing.

Optimization and Some Traditional methods: Introduction to Optimization, Traditional methods of optimization. Some Applications of Soft Computing.

**UNIT II:**

Fuzzy Logic: Introduction to Fuzzy logic. Fuzzy sets and membership functions. Operations on Fuzzy sets. Fuzzy relations, rules, propositions, implications and inferences. Defuzzification techniques. Fuzzy logic controller design. Some applications of Fuzzy logic.

**UNIT III:**

Genetic Algorithms (GA): Concept of "Genetics" and "Evolution" and its application to probabilistic search techniques. Basic GA framework and different GA architectures. GA operators: Encoding, Crossover, Selection, Mutation, etc. Solving single-objective optimization problems using GA. Some specialized GAs: Real-coded GA, Micro-GA

#### UNIT IV:

Multi-objective Optimization problem solving: Concept of multi-objective optimization problems(MOOPs)andissuesofsolvingthem.Multi-ObjectiveEvolutionaryAlgorithm(MOEA). Non-Pareto approaches to solve MOOPs. Pareto-based approaches to solve MOOPs. ome applications with MOEAs.

#### UNIT V:

Artificial Neural Networks: Biological neurons and it's working. Simulation of biological neurons to problem solving. ANN architectures: Different ANN Architectures. ANN Training: Training techniques for ANNs. Applications of ANNs to solve some real-life problems.

#### Text Books

1. R. Rajasekaran and G. A. VijayalakshmiPai, Neural Networks, Fuzzy Logic, and Genetic Algorithms: Synthesis and Applications, Prentice Hall of India, New Delhi, 2003
2. D. K. Pratihar, Soft Computing, Narosa,2008
3. L.Fausett,FundamentalsofNeuralNetworks,PrenticeHall,UpperSaddleRiver,N.J,1994.
4. F.Martin,Mcneill,andEllenThro,FuzzyLogic:APracticalapproach,APProfessional,2000.

#### Reference Books

- J.-S.R.Jang,C.-T.Sun,andE.Mizutani,Neuro-FuzzyandsoftComputing,PHILearning,2009.
- D. E. Goldberg, Genetic Algorithms in Search, Optimisation, and Machine Learning, Addison- Wesley, Reading, MA,1989

PO-CO MAPPING TABLE

PO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	4	2	2	1	1	1	1	1	1	1	1	5
CO2	2	5	4	4	1	1	1	1	1	1	1	4
CO3	2	5	4	4	1	1	1	1	1	1	1	4
CO4	2	5	4	4	1	1	1	1	1	1	1	5
CO5	2	5	4	4	1	1	1	1	1	1	1	4

## **SEM-IV HC-401 APPLIED CRYPTOGRAPHY**

### **Course Outcome**

- CO 1 Illustrate the concepts of Network Security and Compare Various Symmetric and Asymmetric Cryptographic methods used for Network Security
- CO 2 Gain familiarity with prevalent network and distributed system attacks, defences against them, and forensics to investigate the aftermath.
- CO 3 Develop a basic understanding of cryptography, how it has evolved, and some key encryption techniques used today
- CO 4 Summarize different Authentication Techniques such as hashing, various digital signature techniques, etc
- CO 5 Determine appropriate mechanisms for protecting information systems ranging from operating systems to database management systems and to applications.

### **UNIT I**

Introduction, The need of Security, Security approaches, Principles of Security, Types of Security Attacks, Security Services, Security Mechanisms, A model for Network Security.

Cryptography: Concepts and Techniques: Introduction, Plain text and Cipher text, Substitution Techniques, Transposition Techniques, Encryption and Decryption.

Symmetric and Asymmetric Cryptography, Steganography, Key Range and Key Size, Possible types of Attacks.

### **UNIT II**

Symmetric Key Ciphers: Block Cipher Principles and Algorithms. DES, AES, and Blowfish.

Differential and Linear Cryptanalysis, Block Cipher Modes of Operations, Stream Ciphers, RC4, Location and Placement of encryption function, Key Distribution.

Asymmetric Key Ciphers: Principles of Public Key Cryptosystems, Algorithms, RSA, Diffie-Hellman, ECC, Key Distribution.

### **UNIT III**

Message Authentication Algorithms and Hash Function: Authentication Requirements, Functions, Message Authentication Codes, Hash Functions.

Secure Hash Algorithms, Whirlpool, HMAC, CMAC, Digital Signatures.

Authentication Applications: Kerberos, X.509 Authentication Services, Public-Key Infrastructure, Biometric Authentication.

### **UNIT IV**

Web Security: Web Security Considerations, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET). Intruders.

Viruses and Firewalls: Intruders, Intrusion Detection, Password Management, Virus and related threats, Countermeasures, Firewall Design Principles, Types of Firewalls.  
 Case Studies on Cryptography and Security: Secure Inter Branch Transactions, Cross Site Vulnerability, Virtual Elections.

**UNIT V**

Introduction to Information Hiding, Steganography and Watermarking.  
 Fragile watermarking, Reversible watermarking.  
 Importance of digital watermarking, Applications, Properties, Evaluating watermarking systems.

**Text Books**

1. Cryptography and Network Security – AtulKahate –TMH.
2. Data Communications and Networking – BehourzAForouzan

**Reference Book**

1. CyberSecurityOperationsHandbook–J.W.RittiaghouseandWilliamM.Hancock,Elsevier.
2. CryptographyandNetworkSecurityPrinciplesandPractice–W.Stallings,PearsonEducationAsia.

**CO-PO MAPPING TABLE**

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO2	PSO 3	PSO 4
CO 1	5	1	2	1	2	1	1	1	1	1	1	1	1	4	1	4
CO 2	5	2	4	1	1	2	1	2	1	1	1	1	4	5	4	1
CO 3	4	4	2	2	4	2	1	1	1	1	1	1	4	5	4	5
CO 4	4	4	4	2	2	1	1	2	1	1	1	1	1	4	4	5
CO 5	4	2	5	1	1	4	1	4	1	1	1	1	5	1	5	1

(a)

(b)

- a) Java Practical
- b) Applied Cryptography Practical

Learning /Course Outcomes

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

- CO 1: Implement the cipher techniques
- CO 2: Develop the various security algorithms
- CO 3: Use different open source tools for network security and analysis
- CO 4: To Understand OOP concepts and basics of Java programming**

**List of Experiments**

a) **Java Programming**

1. WAP to display Fibonacci series upto n terms terms
2. WAP to demonstrate String class and its methods.
3. Program to demonstrate use of class and objects.
4. Write a java program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.
5. Program to demonstrate use of inheritance
6. Program to demonstrate use of exception handling (Divide by Zero & Array out of Bound)
- 7.

b) **Applied Cryptography**

Implement following Cryptography algorithm using JAVA

1. Caesar Cipher
2. Play fair Cipher
3. Vigenere Cipher.
4. Diffie-Hellman Key exchange
5. DES
6. AES

CO-PO Mapping table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	4	4	1	1	1	1	1	1	1	5
CO2	2	4	4	4	5	1	1	1	1	1	1	5
CO3	2	4	2	2	5	1	1	1	1	1	1	5
CO4	2	2	2	2	1	1	1	1	1	1	1	5

## **SEM-IV CE-401 DATA MINING**

### **Course Outcome**

CO 1: Design a data mart or data warehouse for any organization

CO 2: Develop skills to write queries using DMQL

CO 3: Extract knowledge using data mining techniques

CO 4: Adapt to new data mining tools.

CO 5: Apply the techniques of clustering, classification, association finding, feature selection and visualization to real world data

### **UNIT I:**

Introduction to data mining: Motivation, Importance, Definition of Data Mining, Kind of Data, Data Mining Functionalities, Kinds of Patterns, Classification of Data Mining Systems, Data Mining Task Primitives, Integration of A Data Mining System with A Database or Data Warehouse System, Major Issues in Data Mining

### **UNIT II:**

Types of Data Sets and Attribute Values, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity. PREPROCESSING: Data Quality, Major Tasks in Data Preprocessing, Data Reduction, Data Transformation and Data Discretization, Data Cleaning and Data Integration. Data warehousing and on-line analytical processing: Data Warehouse basic concepts, Data Warehouse Modeling

### **UNIT III:**

Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute-Oriented Induction. Data Cube Technology: Efficient Methods for Data Cube Computation, Exploration and Discovery in Multidimensional Databases.

### **UNIT IV:**

Mining frequent patterns, associations and correlations: Basic Concepts, Efficient and Scalable Frequent Item set Mining Methods, Are All the Pattern Interesting, Pattern Evaluation Methods, Applications of frequent pattern and associations. Frequent pattern and association mining: A Road Map, Mining Various Kinds of Association Rules, Constraint-Based Frequent Pattern Mining, Extended Applications of Frequent Patterns.

## UNIT V:

Classification: Basic Concepts, Decision Tree Induction, Bayesian Classification Methods, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy: Ensemble Methods, Handling Different Kinds of Cases in Classification, Bayesian Belief Networks, Classification by Neural Networks, Support Vector Machines, Pattern-Based Classification, Lazy Learners (or Learning from Your Neighbors), Other Classification Methods.

### Text Books

1. Jiawei Han, Micheline Kamber, Jian Pei (2012), Data Mining: Concepts and Techniques, 3rd. edition, Elsevier, United States of America.

### Reference Books:

1. Margaret H Dunham (2006), Data Mining Introductory and Advanced Topics, 2<sup>nd</sup> edition, Pearson Education, New Delhi, India.
2. Amitesh Sinha (2007), Data Warehousing, Thomson Learning, India.

## PO-CO MAPPING TABLE

### SEM-IV CE-401 INTERNET OF THINGS

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	4	4	1	1	1	1	1	1	1	2
CO2	2	4	5	5	1	1	1	1	1	1	1	2
CO3	2	4	2	2	1	1	1	1	1	1	1	2
CO4	2	4	4	4	1	1	1	1	1	1	1	2
CO 5	2	4	4	4	1	1	1	1	1	1	1	2

### Course Outcome

- CO1: Identify the Components that forms part of IoT Architecture.
- CO 2: Determine the most appropriate IoT Devices and Sensors based on Case Studies.
- CO 3: Setup the connections between the Devices and Sensors.
- CO 4: Evaluate the appropriate protocol for communication between IoT.
- CO 5: Analyse the communication protocols for IoT.



## **UNIT I**

Fundamentals of IoT: Evolution of Internet of Things, Enabling Technologies, M2M Communication, IoT World Forum (Io TWF) standardized architecture, Simplified IoT Architecture, Core IoT Functional Stack, Fog, Edge and Cloud in IoT Functional blocks of an IoT ecosystem, Sensors, Actuators, Smart Objects and Connecting Smart Objects.

## **UNIT II**

Io T Protocols: Io T Access Technologies: Physical and MAC layers, topology and Security of IEEE 802.15.4, 802.11ah and Lora WAN, Network Layer: IP versions, Constrained Nodes and Constrained Networks,6LoWPAN Application Transport Methods: SCADA, Application Layer Protocols: CoAP and MQTT.

## **UNIT III**

Design and Development: Design Methodology, Embedded computing logic microcontroller, System on Chips, IoT system building blocks IoT Platform overview: Overview of IoT supported Hardware platforms, Raspberry pi, Arduino Board details

## **UNIT IV**

Data Analytics and Supporting Services: Introduction, Structured Versus Unstructured Data Data in Motion vs Data at Rest, Io T Data Analytics Challenges, Data Acquiring Organizing in Io T/M2M,Supporting Services: Computing Using a Cloud Platform for IoT/M2M

## **UNIT V**

Applications/Services, Everything as a service and Cloud Service Models. Case Studies/Industrial Applications: IoT applications in home, infrastructures, buildings, security, Industries, Home appliances. Other IoT electronic equipment, Industry 4.0 concepts.

### **Text Books**

1. IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, David Hanes,GonzaloSalgueiro,PatrickGrossetete,RobBartonandJeromeHenry,CiscoPress,2017.
2. Internet of Things: Architecture, Design Principles and Applications, Rajkamal, McGraw Hill Higher Education

**Reference Books**

1. The Internet of Things – Key applications and Protocols, Olivier Hersent, David Boswarthick, Omar Elloumi and Wiley, 2012 (forUnit2).
2. Recipes to Begin, Expand, and Enhance Your Projects, 2nd Edition, Michael Margolis, Arduino
3. Cookbook and O’Reilly Media,2011

<b>CO1</b>	2	2	4	4	1	1	1	1	1	1	1	2
<b>CO2</b>	2	2	4	4	1	1	1	1	1	1	1	2
<b>CO3</b>	2	4	5	5	1	1	1	1	1	1	1	2
<b>CO4</b>	2	4	2	2	1	1	1	1	1	1	1	2
<b>CO 5</b>	2	4	4	4	1	1	1	1	1	1	1	2

PO-CO MAPPING TABLE

# DEPARTMENT OF COMPUTER SCIENCE

## SYLLABUS OF Ph.D. PROGRAMME



**RAMA DEVI WOMEN'S UNIVERSITY**  
Vidya Vihar, Bhubaneswar-751022, Odisha  
Website: <https://rdwu.ac.in>

# Ph. D. COURSE WORK SYLLABUS

(2021)



*Masterki*  
21.10.23  
Controller of Examinations  
R.D. Women's University  
Bhubaneswar

**P. G. DEPARTMENT OF COMPUTER SCIENCE  
RAMA DEVI WOMEN'S UNIVERSITY  
VIDYA VIHAR, BHUBANESWAR-751022**

## SYLLABUS FOR Ph.D. COURSE-WORK

All the scholars of Ph.D. programme of this Department have to undergo a course work of one semester duration. On successful completion of the course work, the scholars may be eligible for registration for Ph.D. subject to final recommendation of Department Research Committee (DRC). The syllabus structure for coursework is given below.

Papers	Course Title	Credits	Marks	Pass Mark	Remarks	Page No.
Paper-I	Research Methodology and Computer Application (Theory & Practical)	4	100	50%	Subject Specific	
Paper-II (Elective)	<b>Elective:</b> <i>Subject Specific</i> (Theory)	4	100	50%	Subject Specific	
Paper-III	Review of Related Literature (Practical)	4	100	50%	Common to all subjects	
Paper-IV	Research and Publication Ethics (Theory & Practical)	4	100	50%	Common to all subjects	
--	<b>Total</b>	<b>16</b>	<b>400</b>	<b>50%</b>	--	--



## PAPER-I

FM-100 (Theory-75 + Practical-25)

### RESEARCH METHODOLOGY AND COMPUTER APPLICATIONS

Course Code: CS-01

Credit 4

#### Unit-I: Philosophy and Culture of Research

Introduction to Philosophy of Knowledge: Various aspects of research methodology; Methodological Approaches; the Analytical Approach; Methodological Procedures with Integrated Approaches.

Types of Research: Introduction to Qualitative & Quantitative Research- Need for Qualitative & Quantitative Research, Grounded Theory Approach, Case Studies in Qualitative Research,

The Challenges of Reliability and Validity in Qualitative Research.

Hypothesis: Different types, Significant, Development of working hypothesis, Null hypothesis.

#### Unit-II: Scientific Writing and Presentation Tools

Structure of a Research Manuscript: Paper Title, Abstract, Introduction, Citations, Review of Literature, Identifying/ narrow-down to Problem Statements, Proposed Methods/Problems, Result Discussion, Conclusion and Future Directions, References Styles and Citation Credits. Structure of Dissertation Preparation. Presentation Tools (LaTeX), Open Source Tools for Figure Preparations.

#### Unit-III: Data Analysis in Research

Basic statistical methods through the use of linear model theory and regression. Analysis through one-end two-sample t-tests, multiple linear regression, analysis of variance, regression diagnostics, model-building techniques, random effects models, mixed models. Need of Quantitative analysis in Research Methodology, Data collection, Univariate & Bi-variate data analysis and application to various projects.

Procedure for data collection and data analysis techniques using Python.

#### Unit-IV: Practical on Computer Applications

Approaches to Computer Application: MS Word: Working with Text, Working with Tables, Graphics and Pages, Document Views and Formatting, and Mail-merge, and Referencing Style. MS-Office and its application, File handing in window, various versions of MSOffice, Research publishing tool- MS-Word, Adobe acrobat, Graphics tool- MS Excel, MS-Power Point: Creating presentations and adding effects, Subject/field specific tools on [www.freeware.com](http://www.freeware.com) Use of Internet: Fundamentals and Services – E-mail, FTP, Telnet, WWW

#### Text Books:

1. Research Methodology: Methods and Techniques, Second Edition, New Age International publishers by C.R. Kothari
2. Mu current presentation tools by Michel Hyatt, International Leadership Groups
3. Morgan, G. and Smircich, L., The Case for Qualitative Research, Academy of Management Review, 5(4): 491-500.
4. 'The Craft of Research' by Wyne C. Booth, Colomb, William, University of Chicago Press. <https://press.uchicago.edu/ucp/books/book/chicago/C/bo23521678.html>

**Unit I: Cloud Computing and Networking**

Cloud Computing: Virtual Machine Management: Configuration, Placement and Resource Allocation. Creating and Configuring Hyper-V Network Virtualization, Overview of Backup and Restore Options for Virtual Machines, Protecting Virtualization Infrastructure by Using Data Protection Manager. Power efficiency in Virtual Data centres, Fault Tolerance in Virtual Data Centres. Term Papers [as provided].

Networking: Layers and Functions, Switching techniques, Addressing, Routing Protocols, Quality of Services, Network Virtualization, Recent research trends, Term Papers [as provided].

Concept implementations using NS2/NS3.

**Unit II: Soft Computing**

Artificial Neural Networks: Introduction to Artificial Neural Networks (ANNs), ANN Architectures, Training techniques for ANNs. Single-Layer and multi-layer NN System, Back Propagation Network: Multi-layer feed-forward network, training using back propagation algorithm, Recurrent Neural Networks (RNNs): Elman Network, Jordan Network, The Hopfield network, Boltzmann machines, Convolutional Networks.

Implementations using Scilab/Matlab/Open Source Software.

**Unit III: Advanced Algorithm**

Linear Programming: Introduction, Example, Formal Structure, Standard Form, Slack Form, Simplex Algorithm, Initialise Simplex, Duality.

Flow Networks: Max Flow Problem, Min Cut Problem, Equivalence of Max Flow and Min-Cut, Residual Graph, Cut, Idea of Ford Fulkerson, Augmenting Path & Ford Fulkerson.

NP-Completeness: Meaning of P and Polynomial time, Polynomial-Time Reductions (Vertex Cover to/from Independent Set, Vertex Cover to Set Cover), Polynomial-Time Verification, NP, NP-Completeness (Example Problems: 3-SAT, Vertex Cover), Reductions (Independent Set to Vertex Cover, Vertex Cover to Set Cover).

**Unit IV: Applied Cryptography**

Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques, Transportation Techniques.

Introduction to Information Hiding, Steganography and Watermarking, Fragile watermarking, Reversible watermarking, Importance of digital watermarking, Applications, Properties, Evaluating watermarking systems.

Implementations using Scilab/Matlab.

## Text Books:

1. Computer Networking *A Top Down Approach* by James F. Kurose and Keith W. Ross, 6<sup>th</sup> Edition, Pearson.
2. Cloud Computing: Concepts, Technology & Architecture, Thomas Erl, Ricardo Puttini, Zaigham Mahmood, Prentice Hall.
3. B Kröse, P Van Der Smagt - An Introduction to Neural Networks, University of Amsterdam, 8<sup>th</sup> ed., 1996
4. Simon Haykin, Neural Networks and Learning Machines, (3<sup>rd</sup>Edn.), PHI Learning, 2011.
5. Algorithm Design, Jon Kleinberg, Eva Tardos.
6. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson and Ronald L. Rivest.
7. Approximation Algorithms: Vijay V.Vazirani
8. W.Stallings- Cryptography and Network Security Principles and Practice, Person Education Asia, 2000. (3rd Edition).
9. Ingemar J. Cox, Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, Ton Kalker, "Digital Watermarking and Steganography", Morgan Kaufmann Publishers, New York, 2008.





## PAPER- III: REVIEW OF RELATED LITERATURE

Course Code: CS-03

Credit: 04

**Full Marks: 100 (Practical)**

### Learning Outcomes:

After completion of the course the students will be able to-

- Conduct review of related literature
- Identify the research gap and write the review in a synchronized manner
- Select a research area of their interest
- Identify variables relevant to the selected research area
- Summarize the findings of different research studies
- Write a thematic paper on any contemporary issue in the subject
- Present thematic paper

### CONTENTS

Each student is required to select a problem on which she has to do intensive review of related studies under the supervision of a faculty member of the Department. She has to review adequate research studies related to the problem and prepare a report.

The student is required to submit a report on the review carried out by her and need to give a power point presentation before the RAC. Assessment shall be made on the basis the following criteria:

- 1) Relevance of the reviews.
- 2) Finding the research gap.
- 3) Standard and quality of writing the review.
- 4) Style of presentation.
- 5) Answering the question

### Distribution of Marks for Evaluation

1) Report writing and submission	: 50 Marks
2) Presentation	: 30 Marks
3) Viva-voce Test	: 20 Marks
-----	
Total:	100 Marks

\*\*\*

## PAPER- IV: RESEARCH AND PUBLICATION ETHICS

Course Code: CS-04

Credits: 04

Full Marks: 100 (Theory-75) + (Practical-25)

### BACKGROUND

This Paper has been incorporated in the Ph.D. course work under this University as per the UGC correspondence in December 2019 vide its 543rd Meeting held on 09 August 2019. The main aim of this course is to create awareness about publication ethics and publication misconducts. It is an interdisciplinary course with 03 credits (Theory) and 1 credit (Practical). The course transaction modes shall be classroom teaching, guest lectures, group discussions and practical sessions. The evaluation of the course will be through continuous assessment processes such as tutorials, assignments quizzes, etc.. Final examination will be conducted at the end of the course.

### LEARNING OUTCOMES:

On completion of the course, the scholars will be able to:

- i. Understand the basics of philosophy of science and ethics, research integrity, publication ethics.
- ii. Identify research misconduct and predatory publications.
- iii. Comprehend indexing and citations, open access publications, research metrics (citations, h-index, impact factor etc).
- iv. Use plagiarism tools for a valid and ethical research report.

### COURSE STRUCTURE:

#### A: THEORY

#### Unit-I: Philosophy and Ethics

- Introduction to Philosophy: Definition, nature and scope, concept, branches.
- Ethics: definition, moral philosophy, nature of moral judgement and reactions.
- Intellectual honesty and research integrity
- Conflict interest

#### Unit-II: Scientific Conduct

- Ethics with respect to science and research
- Scientific misconduct: Falsification, Fabrication, and Plagiarism (FFP)
- Redundant Publications: Duplicate and overlapping publications.
- Selective reporting and misrepresentation of data.

#### Unit-III: Publication Ethics

- Publication ethics: Definition, introduction and importance
- Violation of publication ethics, authorship and contributorship
- Publication misconduct: Definition, concept, problems that lead to unethical behaviour, types, identification of publication misconduct, complaints and appeals
- Predatory publishers and journals

**Unit-I: Open Access Publishing**

- Open Access Publications and initiatives
- Online resource to check publisher copyright and self-achieving policies (SHERPA/RoMEO)
- Journal finder/ journal suggestion tools viz. Elsevier finder, Springer, Journal suggester etc.

**Unit-II: Publication Misconduct**

- Use of plagiarism software like Turnitin, Urkund and other open source software tools
- Software tools to identify predatory publications developed by SPPU
- Indexing databases

**Unit-III: Database and Research Metrics**

- Citation databases: Web of Science, Scopus. etc.
- Impact Factor of journal as per Journal Citation Report.
- Metrics: h-index, g-index, i10 index, altmetrics

**Unit-IV: Group Discussion/Seminar**

- Subject Specific ethical issues, FFP, authorship
- Conflict of interest
- Complaints and appeals: examples and fraud from India and abroad

**REFERENCES**

- Bird, A. (2006). Philosophy of Science. Rutledge.
- MacIntyre, A. (1967). A short history of ethics. London.
- P.Chaddah (2018). Ethics in competitive Research: Do not get scooped; do not get plagiarised.
- National Academy of Sciences (2009). On being a scientist: A guide to responsible conduct in Research (3<sup>rd</sup> Ed.), National Academics Press.
- Resnik, D.B. (2011). What is ethics in research & why is it important. National Institute of Environmental Health Sciences, 1-10.
- Beall, J. (2102). Predatory publishers are corrupting open access. Nature, 489 (7415), 179-179.
- Indian National Science Academy (INSA). Ethics in science education, research and governance (2019).

\*\*\*

