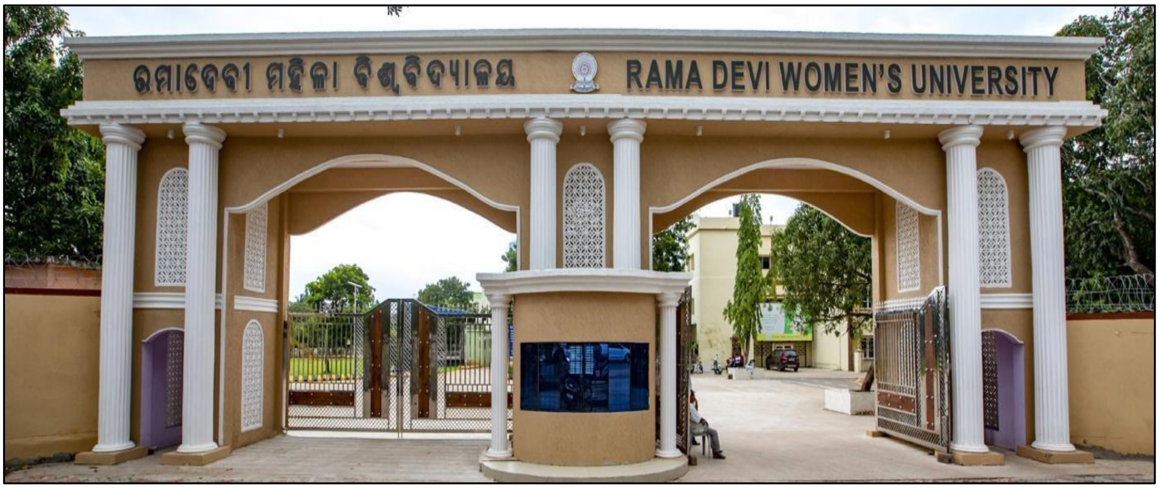
DEPARTMENT OF COMPUTER SCIENCE

SYLLABUS OF UG PROGRAMME (B.Sc.)



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

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)

SEMESTER	COURSE OPTED	COURSE NAME	CREDITS
I	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
II	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
III	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
IV	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
V	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
VI	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 EnhancementCourses(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**disciplineshall choose **four papers of any one discipline** astheir 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 competing 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+……

6. Write a Program to compute the sum of the first n terms of the following series

S =1-2+3-4+5…………….

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 theprime 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 order

    The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

13. Write a Program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
14. Write a program that swaps two numbers using pointers.
15. Write a program in which a function is passed address of two variables and then 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 Recodng 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. BjarneStroustroup, Programming - Principles and Practice using C++,2/e, Addison-Wesley 2014
2. HerbtzSchildt, C++: The Complete reference, MGH,4/ed.
3. P. C. Sethi, P. K. Behera, “Programming in C++”- Kalyani Publisher, Ludhiana

CORE–3Practical: 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! + …………
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 ata 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: Defination, 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 thesemantic and syntax differences between C++ and Java, Compiling andExecuting 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, DoingBasic Program Output, Decision Making Constructs (conditional statements andloops) and Nesting, Java Methods (Defining, Scope, Passing and ReturningArguments, Type Conversion and Type and Checking, Built-in Java ClassMethods). 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 Stringclass, Creating & Using String Objects, Manipulating Strings, String Immutability& Equality, Passing Strings To & From Methods, StringBuffer Classes and StringBuilder Classes. IO package: Understanding StreamsFile class and its methods, Creating, Reading,Writing using classes: Byte and Character streams,FileOutputStream, FileInputStream, FileWriter, FileReader, InputStreamReader, PrintStream, PrintWriter. Compressing and Uncompressing File.

Unit-4

Exception Handling, Threading, Networking and Database Connectivity:Exception types, uncaught exceptions, throw, built-in exceptions, Creating yourown exceptions; Multi-threading: The Thread class and Runnable interface,creating single and multiple threads, Thread prioritization, synchronization andcommunication, suspending/resuming threads. Using java.net package,Overview of TCP/IP and Datagram programming. Accessing and manipulatingdatabases 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 StringBufferclasss like setCharAt( ), setLength(), append(), insert(), concat()and equals().
7. Write a program to create a – “distance” class with methods where distance is computed interms 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 (feetandinches)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 creates a reusable class to generate Fibonacci series, where the function to generate Fibonacci series is given in a different file belonging to the same package.
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 manipuion.

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 ProcessingConcepts 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 1stMonday 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 1stletter 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 thanthe 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

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	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" 7th 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.

<u>CORE – 7 Practical: Discrete Mathematical Structure Lab</u>

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. Morden 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 programusing 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 Cto implement non-preemptive priority based scheduling algorithm.
10. Write a program using Cto implement preemptive priority based scheduling algorithm.
11. Write a program using Cto 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, Slidding 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

CORE – 10: Computer Graphics

COURSE OUTCOMES (CO):

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

CO 1Analyze the background processes involved in computer graphics displays, understanding of Algorithms.
CO 2Use Mathematics in Vector. Create Segments and apply clipping to different shapes.
CO 3Understand and apply algorithms used in Computer Graphics.
CO 4Apply methods suitable for 2D and 3DTransformation such as Translation, Rotation, Scaling, Reflection, Shear etc.
CC 5Apply 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 who lists and tables, CSS ID and class.Box model(introduction, border properties, padding properties, margin properties), CSS colour, groping, Dimensions, display, positioning, floating, align, pseudo class, Navigation bar, image sprites.

Unit-3

Java scripts: Client side scripting, what is java script, simple java script, variables, functions, conditions, loops and repetitions. Java scripts and objects, java script own objects, the DOM and web browser environment, forms and validations.
DHTML: Combining HTML, CSS, java scripts, 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-Willey publishing)
3. John Duckett, Beginning HTML, XHTML, CSS and Java script.

Reference Book:

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

CORE – 11Practical: 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, favoritecolour 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

<u>CORE – 12 Practical: Software Engineering Lab</u>

S. No. Practical Title

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 andfares
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 implementDFS.
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 competing 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 competing 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,LaxmiPublication.
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

DSE – 2:Unix Shell Programming

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 Frontiline" 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", 1stEdition, 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. There 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.

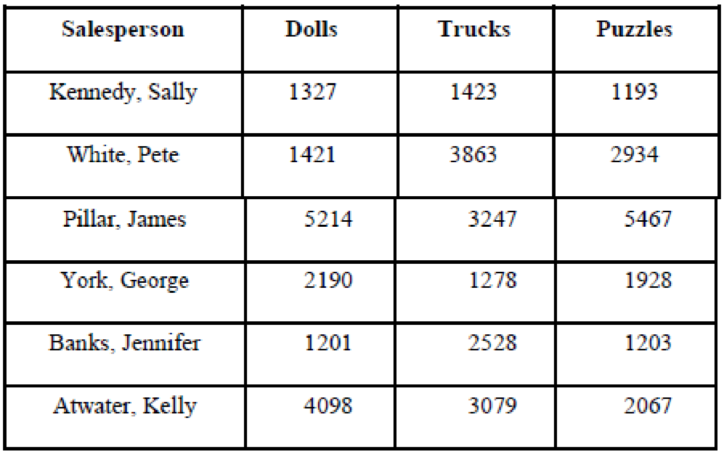
3. Design a time-table form for your college.
   - The first line should mention the name of the college in 16-point Arial Font and should be bold.
   - The second line should give the course name/teacher‘s name and the department in 14-point Arial.

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:

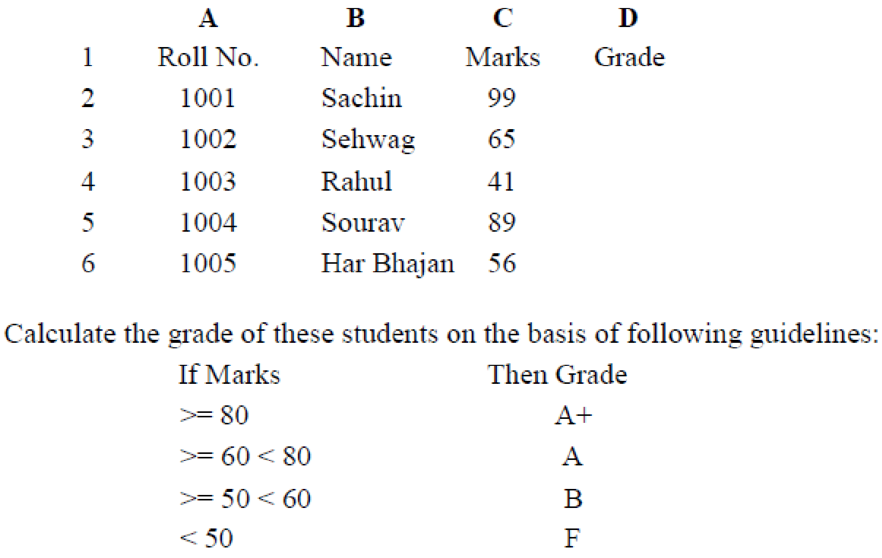


Salesperson	Dolls	Trucks	Puzzles
Kennedy, Sally	1327	1423	1193
White, Pete	1421	3863	2934
Pillar, James	5214	3247	5467
York, George	2190	1278	1928
Banks, Jennifer	1201	2528	1203
Atwater, Kelly	4098	3079	2067

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

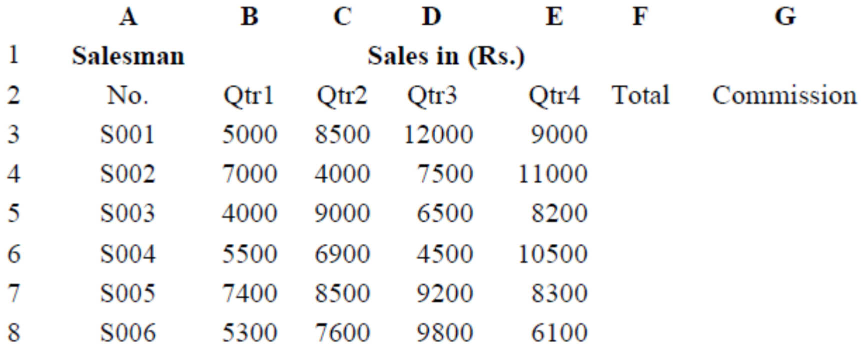


	A	B	C	D
1	Roll No.	Name	Marks	Grade
2	1001	Sachin	99	
3	1002	Sehwag	65	
4	1003	Rahul	41	
5	1004	Sourav	89	
6	1005	Har Bhajan	56	

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

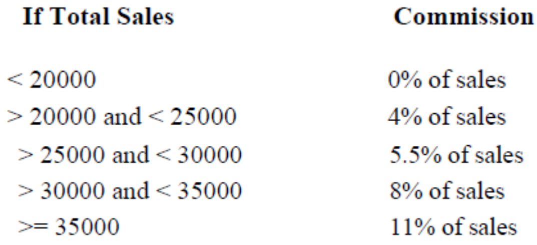
If Marks	Then Grade
>= 80	A+
>= 60 < 80	A
>= 50 < 60	B
< 50	F

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		

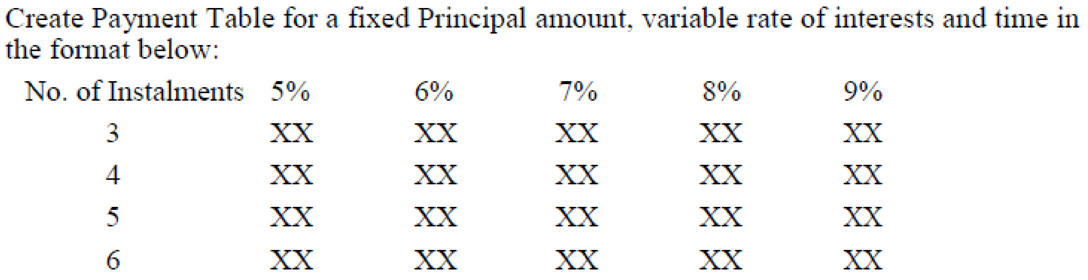
Calculate the commission earned by the salesmen on the basis of following Candidates:



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
>= 35000	11% of sales

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

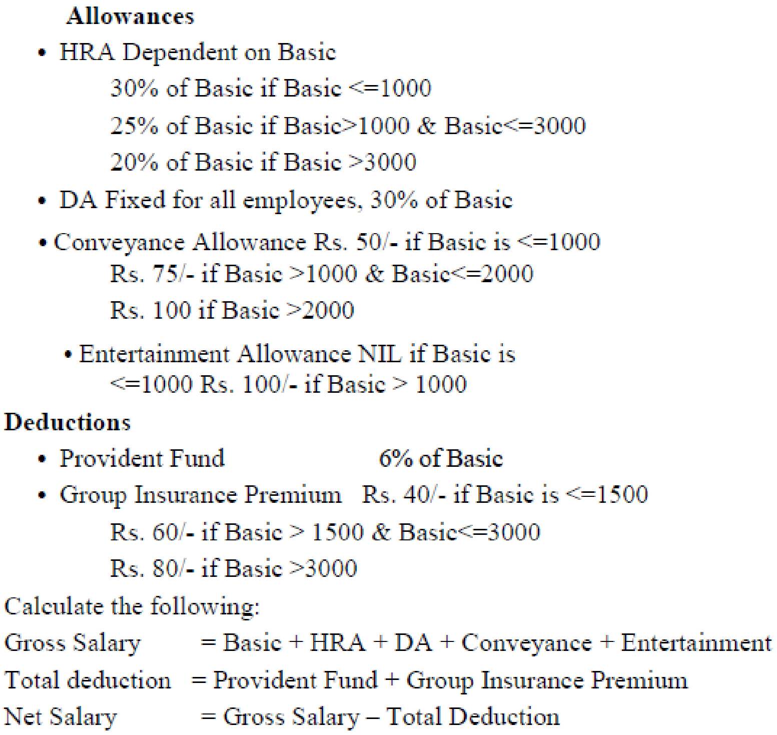
8.



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 <=1000
  25% of Basic if Basic>1000 & Basic<=3000
  20% of Basic if Basic >3000
- DA Fixed for all employees, 30% of Basic
- Conveyance Allowance Rs. 50/- if Basic is <=1000
  Rs. 75/- if Basic >1000 & Basic<=2000
  Rs. 100 if Basic >2000
- Entertainment Allowance NIL if Basic is
  <=1000 Rs. 100/- if Basic > 1000

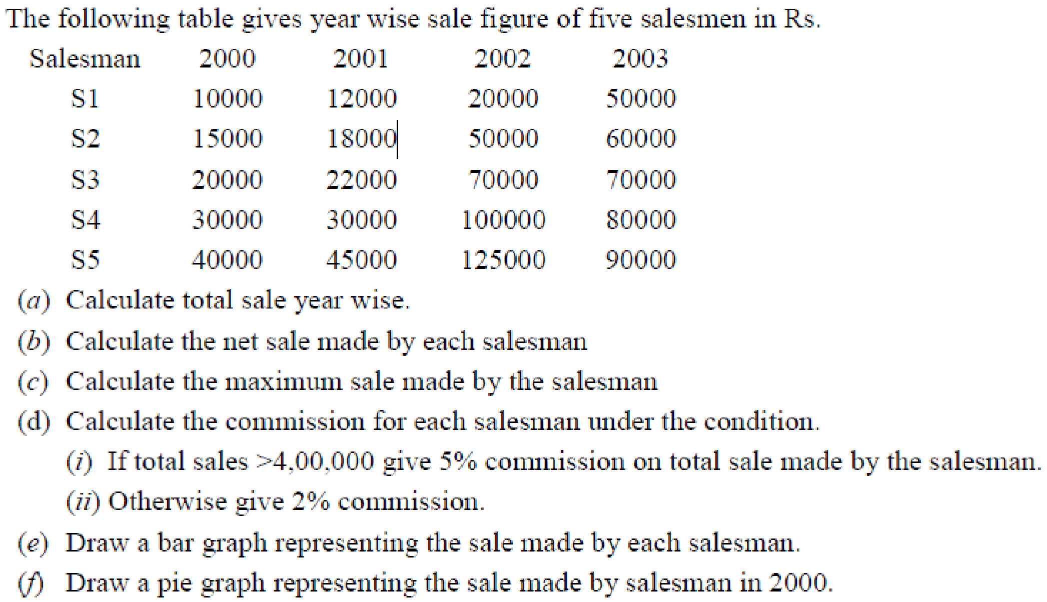
Deductions

- Provident Fund 6% of Basic
- Group Insurance Premium Rs. 40/- if Basic is <=1500
  Rs. 60/- if Basic > 1500 & Basic<=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+……
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.