

B.Sc. Computer Science Honours Course Structure

Semester	Core Courses	Discipline Specific Electives	General Electives	Skill Enhancement Courses	Ability Enhancement Compulsory Courses
I	CC1 CC2		GE1		Environmental Science
II	CC3 CC4		GE2		English/MIL
III	CC5 CC6 CC7		GE3	SEC1	
IV	CC8 CC9 CC10		GE4	SEC2	
V	CC11 CC12	DSE1 DSE2			
VI	CC13 CC14	DSE3 DSE4			

Core Courses (Credit: 6 Each)

CC1 - Programming Fundamentals using C (4 + 2 Lab)
 CC2 - Computer System Architecture (4 + 2 Lab)
 CC3 - Discrete Structures (5 + 1 Tutorial)
 CC4 - Data Structures (4 + 2 Lab)
 CC5 - Operating Systems (5 + 1 Tutorial)
 CC6 - Design and Analysis of Algorithms (4 + 2 Lab)
 CC7 - Programming in JAVA (4 + 2 Lab)
 CC8 - Database Management Systems (4 + 2 Lab)
 CC9 - Software Engineering (5 + 1 Tutorial)
 CC10 - Computer Graphics (4 + 2 Lab)
 CC11 - Theory of Computation (5 + 1 Tutorial)
 CC12 - Computer Networks (5 + 1 Tutorial)
 CC13 - Internet Technologies (4 + 2 Lab)
 CC14 - Artificial Intelligence (5 + 1 Tutorial)

Discipline Specific Electives (Credit: 6 Each)

DSE1 - Numerical Methods (5 + 1 Tutorial)
 DSE2 - Microprocessor (4 + 2 Lab)
 DSE3 - Information Security (5 + 1 Tutorial)
 DSE4 - Cloud Computing (5 + 1 Tutorial)
 DSE5 - Project Work (6)

Choice for Skill Enhancement Courses (Credit: 2 Each)

Semester III: SEC1

Programming in Python

[OR]

Linux Programming

Semester IV: SEC2

Web Technologies

[OR]

Android Programming

General Electives (Credit: 6 Each)

GE1 - Computer Fundamentals

GE2 - Introduction to Programming (using C)

GE3 - Introduction to Database Systems

GE4 - Electronic Commerce Technologies

CORE COURSES

CC1 - Programming Fundamentals using C

Theory: 60 Lectures

1. Overview of C

[7 Lectures]

A Brief History of C, Procedural Programming vs. Object-Oriented Programming, Form of a C Program, Compilation and Execution of a C Program, C's Memory Map.

2. Data Types, Variables, Constants, Operators and Basic I/O

[10 Lectures]

Basic Data Types, Declaration and Definition of Variables, Scope of Variables, Storage Class Specifiers, Variable Initialization, Constants, Type Qualifier (*const*), Operators, Formatted Console I/O (*scanf()*, *printf()*), Unformatted Console I/O (*getchar()*, *putchar()*), Header Files, Use of Comments.

3. Expressions and Statements

[8 Lectures]

Operator Precedence in Expressions, Type Conversion in Expressions, Casts, *True and False in C*, Selection Statements, Iteration Statements, Jump Statements, Block Statements.

4. Arrays, Strings and Functions

[12 Lectures]

One-Dimensional Arrays, Array Initialization, Generating a Pointer to an Array, Passing One-Dimensional Arrays to Functions, Two-Dimensional Arrays, Passing Two-Dimensional Arrays to Functions, Character Arrays, Array of Strings, Indexing Pointers, Introduction to Multi-Dimensional Arrays.

General form of a Function, Function Declaration and Definition, Function Arguments (Call by Value and Call by Reference), Arguments to *main()* (*argc and argv*), Returning from a

Function, Functions Returning Values, Function Returning Pointers, Functions of Type *void*, Declaring Variable-Length Parameter Lists, Recursion, Library Functions.

5. Derived Data Types (Structures and Unions) [5 Lectures]

Basics of Structures, Accessing Structure Members, Structure Assignments, Array of Structures, Pointers to Structures, Passing Structures to Functions, Nested Structures, Self-Referential Structures, Bit-Fields, Unions, *typedef*.

6. Pointers in C [8 Lectures]

Pointer Variables, Pointer Assignments, Pointer/Address Arithmetic, Pointer Comparison, Array of Pointers, Multiple Indirection, Pointers to Functions, Problems with Pointers.

7. Memory Allocation in C [4 Lectures]

Static and Dynamic Memory Allocation, C's Dynamic Allocation Functions (*malloc()*, *calloc()*, *free()*).

8. File I/O and Preprocessor Directives [6 Lectures]

Opening and Closing a File, Reading and Writing Text Files, Random Access in Files, Preprocessor Directives, Macros.

Reference Books:

1. R. Sethi, Programming Languages: concepts and constructs, Addison-Wesley, 1996.
2. B.W. Kernighan and D. M. Ritchie, The C Programming Language, PHI.
3. R.C. Hutchinson and S.B. Just, Programming using the C Language, McGraw-Hill.
4. B.S. Gottfried, Schaum's Outline of Theory and Problems of Programming with C, McGraw-Hill.
5. H. Schildt, C Made Easy, Osborne McGraw-Hill.

Programming Fundamentals using C

Practical:

1. WAP to print the sum and product of digits of an integer.
2. WAP to reverse a number.
3. WAP to compute the sum of the first n terms of the following series
$$S = 1 + 1/2 + 1/3 + 1/4 + \dots$$
4. WAP to compute the sum of the first n terms of the following series
$$S = 1 - 2 + 3 - 4 + 5 - \dots$$
5. 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.
6. Write a function to find whether a given number is prime or not. Use the same to generate the prime numbers less than 100.

7. WAP to compute the factors of a given number.
8. Write a macro that swaps two numbers. WAP to use it.
9. WAP to print a triangle of stars as follows (take number of lines from user):

```
      *
     ***
    *****
   ********
  **********
 **********
```

10. WAP to perform following actions on an array entered by the user:

- i) Print the even-valued elements
- ii) Print the odd-valued elements
- iii) Calculate and print the sum and average of the elements of array
- iv) Print maximum and minimum element of the array
- v) Remove duplicates from the array
- vi) 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.

11. WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

12. Write a program that swaps two numbers using pointers.

13. Write a program in which a function is passed addresses of two variables and then swap the contents.

14. WAP 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.

15. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically.

16. Write a menu driven program to perform the 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 the number of vowels
- i) Reverse the string

17. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.

18. WAP to display Fibonacci Series using (i) Iteration, (ii) Recursion.

19. WAP to calculate Factorial of a number using (i) Iteration, (ii) Recursion.
20. WAP to calculate GCD of two numbers (i) Iteration, (ii) Recursion.
21. Write a menu-driven program to perform the following Matrix operations (2D array implementation):
 - a) Sum
 - b) Difference
 - c) Product
 - d) Transpose
22. Create a structure Student containing fields for Roll-No, Name, Class, Year and Total Marks. Create 10 students and store them in a file.
23. Write a program to retrieve the student information from file created in previous question and print it in following format:
Roll-No Name Marks
24. Copy the contents of one text file to another file, after removing all whitespaces.
25. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.
26. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

CC2 - Computer System Architecture

Theory: 60 Lectures

1. Introduction

[13 Lectures]

Fundamentals of Boolean Algebra, Logic Gates, Switching Functions, Simplification of Switching Functions, Combinational Circuits, Introduction to Sequential Circuits, Flip-Flops, Registers, Counters, Memory Units.

2. Data Representation and Basic Computer Arithmetic

[8 Lectures]

Number Systems and Codes, Complements, Fixed and Floating Point Representation, Character Representation, Addition, Subtraction, Magnitude Comparison, Multiplication and Division Algorithms for Integers.

3. Basic Computer Organization and Design

[10 Lectures]

Computer Registers, Bus System, Instruction Set, Timing and Control, Instruction Cycle, Memory Reference, I/O and Interrupts, Interconnection Structures, Bus Interconnection Design of Basic Computer.

4. Central Processing Unit

[15 Lectures]

Register Organization, Arithmetic and Logical Micro-operations, Stack Organization, Microprogrammed Control, Instruction Formats, Addressing Modes, Instruction Codes,

Machine Language, Assembly Language, RISC and CISC Architectures, Pipelining and Parallel Architecture.

5. Memory Organization [6 Lectures]

Memory Hierarchy, Cache Memory, Associative Memory, Cache Mapping Techniques.

6. Input/Output Organization [8 Lectures]

External Devices, I/O Modules, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access, I/O Channels.

Reference Books:

1. M. Mano, Computer System Architecture, Pearson Education, 1992.
2. A.J. Dos Reis, Assembly Language and Computer Architecture using C++ and JAVA, Course Technology, 2004.
3. W. Stallings, Computer Organization and Architecture Designing for Performance, 8th Edition, Prentice Hall of India, 2009.
4. M.M. Mano, Digital Design, Pearson Education Asia, 2012.
5. Carl Hamacher, Computer Organization, 5th Edition, McGraw Hill, 2012.

Computer System Architecture (Combinational and Sequential Circuits)

Practical:

1. Implement X-OR Gate using NAND Gates.
2. Implement X-OR Gate using NOR Gates.
3. Implement Half-Adder using Basic Gates.
4. Implement Half-Adder using NAND Gates.
5. Implement Full-Adder using NAND Gates.
6. Implement Full-Subtractor using NAND Gates.
7. Implement the Function: $F = (A + \bar{C})(\bar{B} + C)$ using Basic Gates.
8. Implement the Function: $F = (\bar{A}B + AC)$ using Basic Gates.
9. Implement the Function: $F = ABC + DEF$ using IC 7411.
10. Implement a 4-bit Binary Adder using IC 7483.
11. Implement 2×1 Multiplexer using Basic Gates.
12. Implement 4×1 Multiplexer using NAND Gates.
13. Implement 8×1 MUX using IC 74153.
14. Implement a 3-bit Even Parity Checker using Basic Gates.
15. Implement a 2-bit Comparator using Basic Gates.

16. Implement S-R Flip-Flop using NAND Gates.

17. Implement J-K Flip-Flop using NAND Gates.

CC3 - Discrete Structures

Theory: 60 Lectures

1. Introduction

[15 Lectures]

Introduction to Sets, Set Operations, Properties of Sets, Functions, Relations, Properties of Binary Relations, Partial Order Relations, Pigeon-Hole Principle, Principle of Inclusion and Exclusion, Mathematical Induction, Permutation and Combination.

2. Growth of Functions

[8 Lectures]

Asymptotic Notations, Summation Formulas and Properties, Bounding Summations.

3. Recurrences

[10 Lectures]

Recurrence Relations, Generating Functions, Linear Recurrence Relations with Constant Coefficients and their Solutions, Substitution Method, Recursion Trees, Master Theorem.

4. Graph Theory

[15 Lectures]

Introductions to Graphs, Finite and Infinite Graphs, Graph Isomorphism, Walks, Paths and Circuits, Connected and Disconnected Graphs, Euler Graphs, Hamiltonian Paths and Circuits, Directed Graphs, Weighted Graphs, Graph Representation, Planar Graphs, Graph Colouring, Introduction to Trees, Properties of Trees, Spanning Trees.

5. Propositional Logic

[12 Lectures]

Logical Connectives, Tautology and Contradiction, Logical Equivalences, Propositional Functions, Predicates and Quantifiers.

Reference Books:

1. C.L. Liu, D.P. Mahopatra, Elements of Discrete mathematics, 2nd Edition, Tata McGraw Hill, 1985.

2. Kenneth Rosen, Discrete Mathematics and Its Applications, 6th Edition, McGraw Hill 2006.

3. M.O. Albertson and J.P. Hutchinson, Discrete Mathematics with Algorithms, John Wiley Publication, 1988.

4. J.L. Hein, Discrete Structures, Logic, and Computability, 3rd Edition, Jones and Bartlett Publishers, 2009.

5. D.J. Hunter, Essentials of Discrete Mathematics, Jones and Bartlett Publishers, 2008.

CC4 - Data Structures

Theory: 60 Lectures

1. Arrays [3 Lectures]

Single and Multi-Dimensional Arrays, Sparse Matrices (Array and Linked Representation).

2. Linked Lists [7 Lectures]

Singly, Doubly and Circular Linked Lists, Basic Operations on these Lists.

3. Stacks [6 Lectures]

Implementing Stack (Array and Linked Representation), Prefix, Infix and Postfix Expressions, Utility and Conversion of these Expressions from one to another, Applications of Stack, Limitations of Array Representation of Stack.

4. Queues [5 Lectures]

Array and Linked Representation of Queues, Circular Queues, Priority Queues.

5. Recursion [6 Lectures]

Developing Recursive Definitions of Simple Problems and their Implementations, Tracing Recursion, Analysing Recursion, *Towers of Hanoi*.

6. Trees [10 Lectures]

Introduction to Trees, Binary Trees, Tree Traversals, Binary Search Trees, Various operations on Binary Search Trees, Height-Balanced Trees, Various Operations on Adelson-Velski and Landis (AVL) Trees.

7. Searching and Sorting [7 Lectures]

Linear Search, Binary Search, Comparison of Linear Search and Binary Search, Selection Sort, Bubble Sort, Insertion Sort, Shell Sort, Comparison of Sorting Techniques.

8. Hashing [6 Lectures]

Direct Address Table, Introduction to Hashing, Chaining, Open Addressing (Linear Probing, Quadratic Probing, Double Hashing).

Reference Books:

1. Aaron M. Tenenbaum, Moshe J. Augenstein, Yedidyah Langsam, "Data Structures Using C and C++", 2nd Edition, PHI, 2009.
2. Mark Allen Weiss, "Data Structures and Algorithms Analysis in Java", Pearson Education, 3rd Edition, 2011.
3. Aaron M. Tenenbaum, Moshe J. Augenstein, Yedidyah Langsam, "Data Structures Using Java", 2003.
4. John Hubbard, "Data Structures with JAVA", McGraw Hill Education (India) Private Limited, 2nd Edition, 2009.

5. Goodrich, M. and Tamassia, R. “Data Structures and Algorithms Analysis in Java”, 4th Edition, Wiley, 2013.
6. Herbert Schildt, “Java The Complete Reference (English) 9th Edition Paperback”, Tata McGraw Hill, 2014.

Data Structures

Practical:

1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search.
2. WAP to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.
3. Implement Linked Lists. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists.
4. Implement Doubly Linked List. Include functions for insertion, deletion, search of a number and reverse of the list.
5. Implement Circular Linked List. Include functions for insertion, deletion, search of a number in the list and reverse of the list.
6. Perform Stack operations using Array implementation.
7. Perform Stack operations using Linked List implementation.
8. Perform Circular Queue operations using Array implementation.
9. WAP to scan a polynomial using linked list and add two polynomials.
10. WAP to calculate factorial and to compute the factors of a given number using (i) Recursion, (ii) Iteration.
11. WAP to display Fibonacci Series using (i) Recursion, (ii) Iteration.
12. WAP to calculate GCD of two numbers with recursion and without recursion.
14. WAP to create a Binary Search Tree and perform the following operations:
 - (a) Search a key in the BST
 - (b) Insertion
 - (c) Deletion
 - (d) Display height of the tree
 - (e) Count non-leaf and leaf nodes
15. WAP to convert the Sparse Matrix into non-zero form and vice-versa.

CC5 - Operating Systems

Theory: 60 Lectures

1. Introduction

[6 Lectures]

Operating System (User View and System View), Functions of Operating Systems, Types of Operating Systems (Multiprogramming Systems, Batch Systems, Time Sharing Systems, Operating Systems for Personal Computers and Workstations, Real-Time Systems), *System Boot*.

2. Operating System Organization

[5 Lectures]

User Mode and Kernel Mode, Kernels, System Calls and System Programs.

3. Process Management

[25 Lectures]

Concept of a Process, Process States, Process Hierarchy, Threads, Processes vs. Threads, Types of Threads, Multithreading Models, Process Scheduling, Non-preemptive and Preemptive Scheduling Algorithms, Process Synchronization, The Critical-Section Problem, Mutex Locks, Semaphores, Classic Problems of Synchronization, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Avoidance and Detection, Recovery from Deadlock, Inter-Process Communication.

4. Memory Management

[12 Lectures]

Physical and Virtual Address Space, Swapping, Memory Allocation Strategies (Fixed and Variable Partitions), Paging, Segmentation, Virtual Memory, Demand Paging, Page Replacement.

5. Storage Management and I/O Systems

[10 Lectures]

Overview of Mass Storage Structure, Disk Structure, Disk Scheduling, File Concept, File-System Structure, Allocation Methods, I/O Hardware, Polling, Interrupts, Direct Memory Access.

6. Protection and Security

[2 Lectures]

Reference Books:

1. A. Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John Wiley Publications, 2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education, 2007.
3. G. Nutt, Operating Systems: A Modern Perspective, 2nd Edition Pearson Education, 1997.
4. W. Stallings, Operating Systems, Internals & Design Principles , 5th Edition, Prentice Hall of India, 2008.
5. M. Milenkovic, Operating Systems - Concepts and design, Tata McGraw Hill, 1992.

CC6 - Design and Analysis of Algorithms

Theory: 60 Lectures

1. Introduction [5 Lectures]

Analyzing Algorithms, Asymptotic Notations, Time and Space Complexity Analysis of Iterative and Recursive Programs.

2. Algorithm Design Paradigms [15 Lectures]

Divide and Conquer, Dynamic Programming, Greedy Methods.

3. Sorting Algorithms [20 Lectures]

Merge Sort, Quick Sort, Heap Sort, Bucket Sort, Radix Sort, Count Sort, Complexity Analysis of these Algorithms.

4. Lower Bounding Techniques [2 Lectures]

Decision Trees.

5. Balanced Trees [2 Lectures]

Red-Black Trees.

6. Graph Algorithms [10 Lectures]

Breadth First Search, Depth First Search, Applications of BFS and DFS, Minimal Spanning Tree Algorithms.

7. String Processing [6 Lectures]

String Matching, Knuth-Morris-Pratt (KMP) Algorithm.

Reference Books:

1. T.H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein Introduction to Algorithms, 3rd Edition, PHI, 2009.
2. Sarabasse & A.V. Gelder Computer Algorithm - Introduction to Design and Analysis, 3rd Edition, Pearson, 1999.
3. A. Aho, V. Alfred, J. Hopcroft and J. D. Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley.
4. S.E. Goodman and S.T. Hedetniemi, Introduction to the Design and Analysis of Algorithms, McGraw Hill.
5. Anany V. Levitin, Introduction to the Design & Analysis of Algorithms, Addison-Wesley.

Design and Analysis of Algorithms

Practical:

1. Implement Merge Sort.
2. Implement Quick Sort.

3. Implement Heap Sort.
4. Implement Radix Sort.
5. Create a Red-Black Tree and perform the following operations on it:
 - i) Insert a node
 - ii) Delete a node
 - iii) Search for a number and report the colour of the node containing this number

CC7 - Programming in Java

Theory: 60 Lectures

1. Introduction to Java [8 Lectures]

Java Architecture and Features, Compiling and Executing a Java Program, Variables, Constants, Keywords, Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Type Conversion and Type Casting, Decision Making Constructs (Conditional Statements and Loops) and Nesting, Java Methods (Definition, Scope, Passing and Returning Arguments, Built-in Java Class Methods).

2. Arrays, Strings and I/O [15 Lectures]

Creating and Using Arrays (One-Dimensional and Multi-Dimensional), Referencing Arrays Dynamically, The Java *String* Class, Creating and Using String Objects, Manipulating Strings, String Immutability and Equality, Passing Strings to and from Methods, *StringBuffer* Class, Simple I/O using *System.out* and *Scanner* Class, Byte and Character Streams, Reading and Writing from Console and Files.

3. Object-Oriented Programming Overview [8 Lectures]

Principles of Object-Oriented Programming, Defining and Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables and Methods, Objects as Parameters, Final Classes, *Object* Class, Garbage Collection.

4. Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Annotations (Metadata) [14 Lectures]

Inheritance (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, Extending Interfaces and Packages, Package and Class Visibility, Using Standard Java Packages (*util*, *lang*, *io*), Wrapper Classes, Enumerations, Autoboxing and Unboxing, Annotations.

5. Exception Handling, Threading. [15 Lectures]

Exception Types, Uncaught Exceptions, Built-in Exceptions, Creating Your Own Exceptions, The *Thread* Class and *Runnable* Interface, Creating Single and Multiple Threads, Thread Prioritization, Synchronization and Communication, Suspending and Resuming Threads.

Reference Books:

1. Ken Arnold, James Gosling, David Holmes, “The Java Programming Language”, 4th Edition, 2005.
2. James Gosling, Bill Joy, Guy L Steele Jr. Gilad Bracha, Alex Buckley “The Java Language Specification, Java SE 8th Edition (Java Series)”, Published by Addison-Wesley, 2004.
3. Joshua Bloch, Effective Java 2nd Edition, Publisher: Addison-Wesley, 2008.
4. Rodgers Cadenhead, Laura Lemay, Sams Teach Yourself Java 2 in 21 Days, Sams Publishing.
5. E. Balagurusamy, Programming with Java, Tata McGraw Hill.
6. Bruce Eckel, Thinking in Java, Pearson Education.
7. Peter van der Linden, Just Java 2 (Sun Microsystems Press), Prentice Hall.

Programming in Java

Practical:

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 learn use of single dimensional array by defining the array dynamically.
4. To learn use of length in case of a two dimensional array.
5. To convert a decimal to binary number.
6. To check if a number is prime or not, by taking the number as input from the keyboard.
7. 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.
8. Write a program that show working of different functions of *String* and *StringBuffer* class like *setCharAt()*, *setLength()*, *append()*, *insert()*, *concat()* and *equals()*.
9. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions (from lower to higher data type).
10. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword.
11. Write a program to show the use of static functions and to pass variable length arguments in a function.
12. Write a program to demonstrate the concept of boxing and unboxing.
13. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of *Scanner* class in this program).

14. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate Fibonacci series is given in a different file belonging to the same package.
15. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages.
16. Write a program that takes two numbers a and b as input, computes a/b and invokes Arithmetic Exception to generate a message when the denominator is 0.
17. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
18. Write a program to demonstrate priorities among multiple threads.
19. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).

CC8 - Database Management Systems

Theory: 60 Lectures

1. Introduction [6 Lectures]

Traditional File-Based Systems, Characteristics of Database Approach, Data Models, Database System Architecture, Data Independence.

2. Entity-Relationship Modelling [8 Lectures]

Entity Types, Relationships, Constraints, Enhanced Entity-Relationship Model (Generalization, Specialization, Aggregation, Composition).

3. The Relational Model [13 Lectures]

Relational Model Concepts, Relational Constraints, Relational Algebra and Relational Calculus, Data Manipulation Language (DML), Data Definition Language (DDL), SQL Queries, Views.

4. Database Design [15 Lectures]

Mapping ER/EER Model to Relational Database, Purpose of Normalization, Data Redundancy and Update Anomalies, Functional Dependencies, Process of Normalization (Decomposition), Normal Forms, Correctness Criteria for Normalization (Lossless-Join Decomposition and Dependency Preservation).

5. Transaction Processing [8 Lectures]

ACID Properties, Serializability, Transaction States, Concurrency Control Protocols (Lock-Based Protocols), Deadlocks.

6. File Structure and Indexing

[10 Lectures]

Operations on Files, Unordered and Ordered Records, Overview of File Organizations, Indexing Structures for Files (Primary Index, Secondary Index, Clustering Index), Multilevel Indexing using B Trees and B+ Trees.

Reference Books:

1. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education, 2010.
2. R. Ramakrishnan, J. Gehrke, Database Management Systems 3rd Edition, McGraw Hill, 2002.
3. A. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGraw Hill, 2010.
4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education, 2013.

Database Management Systems

Practical:

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 No for each employee with the Employee No 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 No 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 Department 10 and Department 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 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, Salary and Commission for all employees whose commission amount is 14 greater than their salary increased by 5%.
14. Query to display the Current Date.
15. Query to display Name and calculate the Number of Months between today and the date each employee was hired.
16. Query to display the following for each employee:
<Ename> earns <Salary> monthly but wants <3*Current Salary>
Label the column as Dream Salary.
17. Query to display Name, Hire Date and Day of the Week on which the employee started.
18. Query to display Name, Department Name and Department No for all the employees.
19. Query to display unique listing of all Jobs that are in Department 30.
20. Query to display Name, Department Name of all employees who have an 'A' in their names.
21. Query to display Name, Job, Department No and Department Name for all the employees working at the Dallas location.
22. Query to display Name and Employee No along with their Managers' Names and the Managers' Employee No along with the Employee Names who do not have a manager.
23. 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.
24. Query to display the Highest, Lowest, Sum and Average Salaries of all the employees.

25. Query to display the number of employees performing the same Job type functions.
26. Query to display the number of managers without listing their names.
27. Query to display the Department Name, Location Name, No of Employees and the Average Salary for all employees in that department.
28. Query to display Name and Hire Date for all employees in the same Department as Blake.
29. Query to display the Employee No and Name for all employees who earn more than the average salary.
30. Query to display the Names and Salaries of all employees who report to King.
31. Query to display the Department No, Name and job for all employees in the Sales department.

CC9 - Software Engineering

Theory: 60 Lectures

1. Introduction [10 Lectures]

The Evolving Role of Software, Software Characteristics, Changing Nature of Software, Software Engineering as a Layered Technology, Software Process Framework, Framework and Umbrella Activities, Process Models, Capability Maturity Model Integration.

2. Requirement Analysis [10 Lectures]

Software Requirement Analysis, Initiating Requirement Engineering Process, Requirement Analysis.

3. Software Project Management [8 Lectures]

Estimation in Project Planning Process, Project Scheduling.

4. Quality Management [10 Lectures]

Quality Concepts, Software Quality Assurance, Software Reviews, Metrics for Process and Projects.

5. Design Engineering [10 Lectures]

Design Concepts, Architectural Design Elements, Software Architecture, Data Design at the Architecture Level and Component Level, Mapping of Data Flow into Software Architecture, Modelling Component Level Design.

6. Testing Strategies [12 Lectures]

Software Testing Fundamentals, Strategic Approach to Software Testing, Test Strategies for Conventional Software, Validation Testing, System Testing, Black-Box Testing, White-Box Testing and their Types, Basis Path Testing.

Reference Books:

1. R.S. Pressman, Software Engineering: A Practitioner's Approach (7th Edition), McGraw Hill, 2009.
2. P. Jalote, An Integrated Approach to Software Engineering (2nd Edition), Narosa Publishing House, 2003.
3. K.K. Aggarwal and Y. Singh, Software Engineering (2nd Edition), New Age International Publishers, 2008.
4. I. Sommerville, Software Engineering (8th Edition), Addison-Wesley, 2006.
5. D. Bell, Software Engineering for Students (4th Edition), Addison-Wesley, 2005.
6. R. Mall, Fundamentals of Software Engineering (2nd Edition), Prentice Hall of India, 2004.

CC10 - Computer Graphics

Theory: 60 Lectures

1. Introduction [5 Lectures]

Basic Elements of Computer Graphics, Applications of Computer Graphics.

2. Graphics Hardware [8 Lectures]

Architecture of Raster and Random Scan Display Devices, Input/Output Devices.

3. Fundamental Techniques in Graphics [22 Lectures]

Raster Scan Line, Circle and Ellipse Drawing, Thick Primitives, Polygon Filling, Line and Polygon Clipping Algorithms, 2D and 3D Geometric Transformations, 2D and 3D Viewing Transformations (Projections - Parallel and Perspective), Vanishing Points.

4. Geometric Modelling [10 Lectures]

Representing Curves and Surfaces.

5. Visible Surface Determination [5 Lectures]

Hidden Surface Elimination.

6. Surface Rendering [10 Lectures]

Illumination and Shading Models, Basic Colour Models and Computer Animation.

Reference Books:

1. J.D. Foley, A. van Dam, Feiner, Hughes Computer Graphics Principles & Practice 2nd Edition, Publication Addison-Wesley, 1990.
2. D. Hearn, Baker, Computer Graphics, Prentice Hall of India, 2008.
3. D.F. Rogers, Procedural Elements for Computer Graphics, McGraw Hill, 1997.

4. D.F. Rogers, Adams Mathematical Elements for Computer Graphics, McGraw Hill 2nd Edition, 1989.

Computer Graphics

Practical:

1. Write a program to implement Bresenham's line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to apply various 2D transformations on a 2D object (use homogenous coordinates).
6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
7. Write a program to draw Hermite/Bezier curve.

CC11 - Theory of Computation

Theory: 60 Lectures

1. Introduction

[4 Lectures]

Alphabets, Strings, Languages, Operations on Languages (Concatenation, Kleene Closure).

2. Finite Automata and Regular Languages

[20 Lectures]

Regular Expressions, Transition Graphs, Deterministic and Non-Deterministic Finite Automata, NFA to DFA Conversion, Regular Languages and their Relationships with Finite Automata, Pumping Lemma and Closure Properties of Regular Languages.

3. Context-Free Languages

[20 Lectures]

Context-Free Grammars and Languages, Derivations, Parse Trees, Ambiguities in Grammars and Languages, Language Generated by a CFG, Elimination of useless Symbols/Productions, Unit Productions, Chomsky Normal Forms, Pushdown Automata (Deterministic and Non-Deterministic), Pumping Lemma for CFGs, Properties of Context-Free Languages.

4. Turing Machines and Models of Computation

[16 Lectures]

Turing Machine as a Model of Computation, Universal Turing Machine, Language Acceptability, Decidability, Halting Problem, Recursive and Recursively Enumerable Languages, Undecidable Problems.

Reference Books:

1. Introduction to Automata, Formal Languages and Computation, Peter Linz, Narosa Publishing House.

2. Theory of Computer Science, K.L.P. Mishra, PHI, India.
3. Daniel I.A. Cohen, Introduction to computer theory, John Wiley, 1996.
4. Lewis & Papadimitriou, Elements of the Theory of Computation, PHI, 1997.
5. Hopcroft, Aho, Ullman, Introduction to Automata theory, Language & Computation - 3rd Edition, Pearson Education, 2006.

CC12 - Computer Networks

Theory: 60 Lectures

1. Introduction [5 Lectures]

Components of Data Communication, Protocols and Standards, Network Topologies, Layered Network Architecture, OSI Reference Model, Overview of TCP/IP Protocol Suite.

2. Data Communication Fundamentals and Techniques [10 Lectures]

Analog and Digital Signals, Data Rate Limits, Line Coding Schemes, Pulse Code Modulation, Parallel and Serial Transmission, Multiplexing Techniques (FDM, TDM), Transmission Media.

3. Networks Switching Techniques and Access Mechanisms [5 Lectures]

Circuit Switching, Packet Switching (Datagram Networks and Virtual-Circuit Networks).

4. Data Link Layer Functions and Protocols [10 Lectures]

Error Detection and Correction, Data Link Control, Framing and Flow Control, Protocols for Noisy Channels (Stop and Wait ARQ, Go-Back-N ARQ), Point to Point Protocol.

5. Multiple Access Protocols, Ethernet and Connecting Devices [7 Lectures]

Random Access Protocols (CSMA, CSMA/CD), Controlled Access Protocols (Reservation, Polling, Token Passing), IEEE Standard for LANs, 802.3 MAC Frame, Repeaters, Hubs, Switches, Bridges, Routers, Gateways.

6. Networks Layer Functions and Protocols [7 Lectures]

Logical Addressing (Classful and Classless Addressing), Subnetting, Internet Protocol, Routing Algorithms (Distance Vector, Link State).

7. Transport Layer Protocols [10 Lectures]

UDP, TCP, TCP Connection Establishment and TCP Connection Termination, Flow and Congestion Control.

8. Overview of Application Layer Protocols [6 Lectures]

DNS, HTTP, FTP, SMTP and POP.

Reference Books:

1. B.A. Forouzan, Data Communications and Networking, 4th Edition, THM, 2007.
2. A.S. Tanenbaum, Computer Networks, 4th Edition, PHI, 2002.
3. W. Stallings, Data and Computer Communications, Prentice Hall.
4. LL Peterson, BS Davie, Computer Networks: A Systems Approach, 5th Edition, Morgan-Kaufmann, 2011.

CC13 - Internet Technologies

Theory: 60 Lectures

1. Java [2 Lectures]

Use of Objects, Arrays, *ArrayList* Class.

2. JavaScript [15 Lectures]

Data Types, Operators, Functions, Control Structures, Events and Event handling.

3. JDBC [11 Lectures]

JDBC Fundamentals, Establishing Connectivity and Working with Connection Interface, Working with Statements, Creating and Executing SQL Statements, Working with *ResultSet* Objects.

4. Java Server Pages [20 Lectures]

Introduction to JavaServer Pages, HTTP and Servlet Basics, The Problem with Servlets, The Anatomy of a JSP Page, JSP Processing, JSP Application Design with Model View Controller Architecture, Setting Up the JSP Environment, Implicit JSP Objects, Conditional Processing, Displaying Values, Using an Expression to Set an Attribute, Declaring Variables and Methods, Error Handling and Debugging, Sharing Data Between JSP Pages, Requests, and Users, Database Access.

5. Java Beans [12 Lectures]

Java Beans Fundamentals, JAR Files, Introspection, Developing a Simple Bean, Connecting to DB.

Reference Books:

1. Ivan Bayross, Web Enabled Commercial Application Development Using HTML, DHTML, JavaScript, Perl CGI, BPB Publications, 2009.
2. Cay Horstmann, BIG Java, Wiley Publication, 3rd Edition, 2009.
3. Herbert Schildt, Java 7, The Complete Reference, 8th Edition, 2009.
4. Jim Keogh, The Complete Reference J2EE, TMH, 2002.
5. O'Reilly, Java Server Pages, Hans Bergsten, 3rd Edition, 2003.

Internet Technologies

Practical:

Create event-driven programs for the following:

1. Print a table of numbers from 5 to 15 and their squares and cubes using alert.
2. Print the largest of three numbers.
3. Find the factorial of a number n.
4. Enter a list of positive numbers terminated by Zero. Find the sum and average of these numbers.
5. A person deposits Rs. 1000 in a fixed account yielding 5% interest. Compute the amount in the account at the end of each year for n years.
6. Read n numbers. Count the number of negative numbers, positive numbers and zeros in the list.

CC14 - Artificial Intelligence

Theory: 60 Lectures

1. Introduction

[6 Lectures]

Introduction to Artificial Intelligence, Background and Applications, Turing Test and Rational Agent Approaches to AI, Introduction to Intelligent Agents and their Structures, Behaviours and Environments.

2. Problem Solving and Searching Techniques

[20 Lectures]

Problem Characteristics, Production Systems, Control Strategies, Breadth First Search, Depth First Search, Hill Climbing and its Variations, Heuristics Search, Best First Search, A* Algorithm, Constraint Satisfaction Problem, Means-End Analysis, Introduction to Game Playing, Min-Max and Alpha-Beta Pruning Algorithms.

3. Knowledge Representation

[20 Lectures]

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

4. Dealing with Uncertainty and Inconsistency

[8 Lectures]

Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference, Possible World Representations.

5. Understanding Natural Languages

[6 Lectures]

Parsing Techniques, Context-Free and Transformational Grammars, Recursive and Augmented Transition Nets.

Reference Books:

1. DAN.W. Patterson, Introduction to A.I and Expert Systems, PHI, 2007.
2. Russell & Norvig, Artificial Intelligence - A Modern Approach, LPE, Pearson Prentice Hall, 2nd Edition, 2005.
3. Rich & Knight, Artificial Intelligence, Tata McGraw Hill, 2nd Edition, 1991.
4. W.F. Clocksin and Mellish, Programming in Prolog, Narosa Publishing House, 3rd Edition, 2001.
5. Ivan Bratko, Prolog Programming for Artificial Intelligence, Addison-Wesley, Pearson Education, 3rd Edition, 2000.

DISCIPLINE SPECIFIC ELECTIVES**DSE1 - Numerical Methods****Theory: 60 Lectures****1. Errors [4 Lectures]**

Errors in Computation.

2. Interpolation [8 Lectures]

Newton's Forward and Backward Interpolation, Lagrange's Interpolation.

3. Solution of Linear Equation Systems [12 Lectures]

Gauss Elimination Method, Gauss-Seidel Method, Convergence and Errors.

4. Solution of Non-Linear Equation Systems [14 Lectures]

Bisection Method, Regula-Falsi Method, Secant Method, Newton-Raphson Method, Convergence and its Graphical Significance.

5. Solution of Differential Equations [14 Lectures]

Euler Method, Runge-Kutta Method, Second and Fourth Order Method for Solving Differential Equations.

6. Integration [8 Lectures]

Trapezoidal Rule, Simpson's 1/3 Rule.

Reference Books:

1. V. Rajaraman, Computer Oriented Numerical Methods, PHI.
2. F. Acton, Numerical Methods that Work, Harper and Row.
3. S. D. Conte and C.D. Boor, Elementary Numerical Analysis, McGraw Hill.
4. S. S. Shastri, "Introductory Methods of Numerical Analysis", PHI.

5. C. F. Gerald and P.O. Wheatley, Applied Numerical Analysis, Addison-Wesley.

DSE2 - Microprocessor

Theory: 60 Lectures

1. Microprocessor Architecture [15 Lectures]

Internal Architecture, System Bus Architecture, Memory and I/O Interfaces.

2. Microprocessor Programming [20 Lectures]

Register Organization, Instruction Formats, Assembly Language Programming.

3. Interfacing [25 Lectures]

Memory Address Decoding, Cache Memory and Cache Controller, I/O Interfacing, Keyboard Interfacing, Display Interfacing, Timers, Subroutine Calls and Interrupts, Interrupt Controller, DMA Controller, Video Controller, Communication Interfaces.

Reference Books:

1. R. Gaonkar, Microprocessor Architecture, Programming, and Applications with the 8085, Penram.
2. A. Pal, Microprocessors: Principles and Applications, Tata McGraw Hill.
3. Barry B. Brey: The Intel Microprocessors: Architecture, Programming and Interfacing. Pearson Education, Sixth Edition, 2009.

Microprocessor

Practical:

1. Write a program for binary division and multiplication.
2. Write a program for BCD addition and subtraction.
3. Write a program for linear search and binary search.
4. Write a program to add and subtract two arrays.
5. Write a program for binary to ASCII conversion.
6. Write a program for ASCII to binary conversion.

DSE3 - Information Security

Theory: 60 Lectures

1. Introduction [5 Lectures]

Security, Attacks, Computer Criminals, Security Services, Security Mechanisms.

2. Cryptography [12 Lectures]

Substitution Cipher, Transposition Cipher, Confusion and Diffusion, Symmetric and Asymmetric Encryption, Data Encryption Standard, DES Modes of Operation, Uses of Encryption, Hash Function, Key Exchange, Digital Signatures, Digital Certificates.

3. Program Security [8 Lectures]

Secure Programs, Non-Malicious Program Errors, Malicious Codes Viruses, Trap Doors, Salami Attacks, Covert Channels, Control against Program Threats.

4. Protection in OS [9 Lectures]

Memory and Address Protection, Access control, File Protection, User Authentication.

5. Database Security [8 Lectures]

Requirements, Reliability, Integrity, Sensitive Data, Inference, Multilevel Security.

6. Security in Networks [8 Lectures]

Threats in Networks, Security Controls, Firewalls, Intrusion Detection Systems, Secure Emails.

7. Administrating Security [10 Lectures]

Security Planning, Risk Analysis, Organizational Security Policy, Physical Security. Ethical Issues in Security, Protecting Programs and Data, Information and Law.

Reference Books:

1. C.P. Pfleeger, S.L. Pfleeger: Security in Computing, Prentice Hall of India, 2006.
2. W. Stallings: Network Security Essentials: Applications and Standards, 4th Edition, 2010.
3. Thomas R. Peltier, "Information Security Risk Analysis", CRC Press LLC, 2005.

DSE4 - Cloud Computing

Theory: 60 Lectures

1. Overview of Computing Paradigms [8 Lectures]

Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing.

2. Introduction to Cloud Computing [8 Lectures]

History of Cloud Computing, Cloud Service Providers, Benefits and Limitations of Cloud Computing.

3. Cloud Computing Architecture [20 Lectures]

Comparison with Traditional Computing Architecture (Client-Server), Services Provided at Various Levels, Service Models (Infrastructure as a Service (IaaS), Platform as a Service

(PaaS)), How Cloud Computing Works, Deployment Models (Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud).

4. Service Management in Cloud Computing [8 Lectures]

Service Level Agreements (SLAs), Billing and Accounting, Comparing Scaling Hardware (Traditional vs. Cloud), Economics of Scaling.

5. Cloud Security [16 Lectures]

Infrastructure Security (Network Level Security, Host Level Security, Application Level Security), Data Security and Storage (Data Privacy and Security Issues, Jurisdictional Issues Raised by Data Location), Authentication in Cloud Computing.

Reference Books:

1. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010.
2. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley, 2011.
3. Cloud Computing: Principles, Systems and Applications, Editors: Nikos Antonopoulos, Lee Gillam, Springer, 2012.
4. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010.
5. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach, McGraw Hills, 2010.
6. Dimitris N. Chorafas, Cloud Computing Strategies, CRC Press, 2010.

DSE5 - Project Work

SKILL ENHANCEMENT COUSES

SEC1 - Programming in Python

Planning the Computer Program

Concept of Problem Solving, Problem Definition, Program Design, Debugging, Types of Errors in Programming, Documentation.

Techniques of Problem Solving

Flowcharting, Decision Table, Algorithms, Structured Programming Concepts, Programming Methodologies (Top-Down and Bottom-Up Programming).

Overview of Programming

Structure of a Python Program, Elements of Python.

Introduction to Python

Python Interpreter, Using Python as Calculator, Python Shell, Indentation. Atoms, Identifiers and Keywords, Literals, Strings, Operators (Arithmetic, Relational, Logical, Boolean, Assignment, Ternary, Bitwise, Increment, Decrement).

Creating Python Programs

Input and Output Statements, Control Statements (Branching Statements, Looping Statements, Conditional Statements, exit function, Difference among break, continue and pass), Defining Functions, Default Arguments.

Reference Books:

1. T. Budd, Exploring Python, TMH, 1st Edition, 2011.
2. Allen Downey, Jeffrey Elkner, Chris Meyers, How to Think Like a Computer Scientist: Learning with Python
3. <https://docs.python.org/3/>
4. <http://www.ibiblio.org/g2swap/byteofpython/read/>

Programming in Python

Practical:

Section A: Simple Programs

1. Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice.
2. WAP to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria:

Grade A: Percentage ≥ 80
Grade B: Percentage ≥ 70 and < 80
Grade C: Percentage ≥ 60 and < 70
Grade D: Percentage ≥ 40 and < 60
Grade E: Percentage < 40
3. Write a menu-driven program, using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.
4. WAP to display the first n terms of Fibonacci series.
5. WAP to find factorial of the given number.
6. WAP to find sum of the following series for n terms: $1 - 2/2! + 3/3! - \dots n/n!$
7. WAP to calculate the sum and product of two compatible matrices.

Section B: Visual Programs

All the programs should be written using user defined functions, wherever possible.

1. Write a menu-driven program to create mathematical 3D objects:
 - i) Curve
 - ii) Sphere
 - iii) Cone
 - iv) Arrow
 - v) Ring
 - vi) Cylinder
2. WAP to read n integers and display them as a histogram.
3. WAP to display sine, cosine, polynomial and exponential curves.
4. WAP to plot a graph of people with pulse rate p vs. height h. The values of p and h are to be entered by the user.
5. WAP to calculate the mass m in a chemical reaction. The mass m (in gms) disintegrates according to the formula $m = 60/(t+2)$, where t is the time in hours. Sketch a graph for t vs. m, where $t \geq 0$.
6. A population of 1000 bacteria is introduced into a nutrient medium. The population p grows as follows:
$$P(t) = (15000(1+t))/(15+e)$$
where the time t is measured in hours. WAP to determine the size of the population at given time t and plot a graph for P vs. t for the specified time interval.
7. Input initial velocity and acceleration, and plot the following graphs depicting equations of motion:
 - i) velocity wrt time ($v = u + at$)
 - ii) distance wrt time ($s = u*t + 0.5*a*t*t$)
 - iii) distance wrt velocity ($s = (v*v-u*u)/2*a$)

SEC2 - Linux Programming

Linux Basics

Introduction to Linux, Difference between Linux and other Operating Systems, Linux Architecture, Linux File System, Linux Basic Commands, Linux Users and Groups, Permissions for File, Directory and Users, Searching Files and Directories.

Shell Scripting

Shell Definition, Types of Shells, Linux Editors, Modes of Operations in vi Editor, Writing and Executing Shell Scripts, Shell Variables (User-defined and System Variables), Decision Making in Shell Scripts, Loops, Functions, Pipes and Filters, Utilities (*cut*, *paste*, *join*, *tr*, *uniq*), Pattern Matching (*grep*).

Reference Books:

1. Sumitabha, Das, Unix Concepts and Applications, Tata McGraw-Hill Education, 2006.
2. Michael Jang RHCSA/ RHCE Red Hat Linux Certification: Exams (Ex200 & Ex300), 6th Edition, (Certification Press), 2011.
3. Nemeth Synder & Hein, Linux Administration Handbook, Pearson Education, 2nd Edition, 2010.
4. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Unix Network Programming, The Sockets Networking API, Volume 1, 3rd Edition, 2014.

Linux Programming

Practical:

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 display date in the mm/dd/yy format.
3. Write a shell script to display the multiplication table any number.
4. Write a shell script to find the sum of digits of a given number.
5. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
6. Write a shell script to find the GCD of two numbers.
7. Write a shell script to perform the tasks of basic calculator.
8. Write a shell script to find the power of a given number.
9. Write a shell script to find the permutation $P(n, x)$.
10. Write a shell script to find the greatest number among the three numbers.
11. Write a shell script to find the factorial of a given number.
12. Write a shell script to check whether the number is Armstrong or not.
13. Write a shell script to check whether the file have all the permissions or not.
14. Write a shell script to modify *cal* command to display calendars of the specified months.
15. Write a shell script to accept a login name. If it is not a valid login name then display the message: Entered login name is invalid!

SEC3 - Web Technologies

Section A: HTML Programming

Introduction

Introduction to HTML, A simple HTML Document, HTML Tags, HTML Page Structure.

Basics of HTML

Head, Body, Colours, Attributes, Lists (Unordered and Ordered).

Links

Introduction to Links, Relative and Absolute Links, Link Attributes, ID Attribute.

Images

Putting an Image on a Page, Using Images as Links, Putting an Image in the Background.

Tables

Creating a Table, Table Headers, Captions, Spanning Columns, Styling Tables.

Forms

Form Elements, Input Types, Input Attributes, Styling Forms with CSS.

Section B: PHP Programming

Introduction to PHP

History of PHP, PHP with other Technologies, Scope of PHP.

Basic Syntax, Variables and Constants, Data Types, Expressions, Scope of a Variable, Operators, Operator Precedence and Associativity.

Handling HTML Form with PHP

Capturing Form Data, GET and POST Form Methods, Dealing with Multi-value Fields, Redirecting a Form after Submission.

Conditional Events and Loops

Conditional Statements (*if*, *if . . . else*, *if . . . elseif . . . else*), *switch* statement, Loops (*while*, *for*), *goto*, *break*, *continue*, *exit*.

Functions

Need of Functions, Declaration and Calling of Functions, Functions with Arguments, Default Arguments in Functions, Function Arguments with Call by Value and Call by Reference, Scope of a Function.

String Manipulation and Regular Expression

Creating and Accessing Strings, Searching and Replacing Strings, Formatting, Joining and Splitting Strings, String related Library Functions, Use and Advantage of Regular Expressions over in-built Functions, *preg_match()*, *preg_replace()*, *preg_split()* functions in Regular Expressions.

Arrays

Anatomy of an Array, Creating Index-based and Associative Arrays, Accessing Arrays, Looping with Index-based Array and Associative Array (*each()* and *foreach()*).

Database Connection with MySQL

Data Insertion, Extraction, Deletion and Update.

Reference Books:

1. Virginia DeBolt, Integrated HTML and CSS A Smarter, Faster Way to Learn Wiley/Sybex, 2006.
2. Cassidy Williams, Camryn Williams, Introduction to HTML and CSS, O'Reilly, 2015.
3. Steven Holzner, "PHP: The Complete Reference Paperback", McGraw Hill Education (India), 2007.
4. David Sklar, Adam Trachtenberg, "PHP Cookbook: Solutions & Examples for PHP Programmers", 2014.

SEC4 - Android Programming

Introduction

History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture.

Overview of Object Oriented Programming using Java

Basic Concepts of OOP, Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.

Development Tools

Installing and using Eclipse with ADT plug-in, Installing Virtual Machine for Android Sandwich/Jelly Bean (Emulator), Configuring the Installed Tools, Creating an Android Project "Hello Word", Run on Emulator, Deploy it on USB-connected Android Device.

User Interface Architecture

Application Context, Intents, Activity Life Cycle, Multiple Screen Sizes.

User Interface Design

Form widgets, Text Fields, Layouts, Button Control, Toggle Buttons, Spinners (Combo Boxes), Images, Menu, Dialog.

Database

Understanding of SQLite Database, Connecting with the Database.

Reference Books:

1. James C. Sheusi, Android Application Development for Java Programmers, Cengage Learning, 2013.

Online Resources:

1. <http://www.developer.android.com>
2. <http://developer.android.com/about/versions/index.html>
3. <http://developer.android.com/training/basics/firstapp/index.html>
4. <http://docs.oracle.com/javase/tutorial/index.htm> (Available in the form of free downloadable eBooks also)
5. <http://developer.android.com/guide/components/activities.html>
6. <http://developer.android.com/guide/components/fundamentals.html>
7. <http://developer.android.com/guide/components/intents-filters.html>.
8. <http://developer.android.com/training/multiscreen/screensizes.html>
9. <http://developer.android.com/guide/topics/ui/controls.html>
10. <http://developer.android.com/guide/topics/ui/declaring-layout.html>
11. <http://developer.android.com/training/basics/data-storage/databases.html>

Android Programming

Practical:

1. Create “Hello World” Application that will display “Hello World” in the middle of the screen in the emulator. Also display “Hello World” in the middle of the screen in the Android Phone.
2. Create an application with login module (Check username and password).
3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, image will change.
4. Create a menu with 5 options and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher- in-charge of that course should appear at the bottom of the screen.
6. Create an application with three option buttons, on selecting a button colour of the screen will change.
7. Create a Login Application as above (on successful login, pop up a message).
8. Create an application to Create, Insert, update, Delete and Retrieve operation on the database.

GENERAL ELECTIVES

GE1 - Computer Fundamentals

Theory: 60 Lectures

1. Introduction [6 Lectures]

Introduction to Computer Systems, Classification of Computers, Computer Generations, Components of a Computer System, Applications of Computers.

2. Data Representation [12 Lectures]

Number Systems, Conversion from one Number System to another, Binary Arithmetic, Complements, Binary Coding System.

3. Human-Computer Interface [6 Lectures]

Types of Software (System Software and Application Software).

3. Devices [10 Lectures]

Input and Output Devices (with Connections and Practical Demo), Keyboard, Mouse, Joystick, Scanner, OCR, OMR, Barcode Reader, Web Camera, Monitor, Printer, Plotter.

4. Memory [6 Lectures]

Primary, Secondary and Auxiliary Memory, RAM, ROM, Cache Memory, Hard Disks, Optical Disks.

5. Computer Organization and Architecture [12 Lectures]

CPU, Registers, System Bus, Main Memory Unit, Inside a Computer, SMPS, Motherboard, Ports and Interfaces, Expansion Cards, Ribbon Cables, Memory Chips, Processors.

6. Overview of Emerging Technologies [8 Lectures]

Bluetooth, Cloud Computing, Big Data, Data Mining, Mobile Computing, Embedded Systems.

Computer Fundamentals

Practical:

Practical exercises based on MS Office/ Open Office tools using document preparation and spreadsheet handling packages.

MS Word

1. Prepare a Grocery List having four columns (Serial Number, Product Name, Quantity and Price) for the month of April, 06.

- 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. Leave a gap of 12-points.
- The rest of the document should use 10-point Times New Roman font.
- The footer should contain your specifications as the designer and date of creation.

4. BPB Publications plans to release a new book designed as per your syllabus. Design the First Page of the book as per the given specification:

- The title of the book should appear in bold using 20-point Arial font.
- The name of the author and his qualifications should be in the centre of the page in 16-point Arial font.
- At the bottom of the document should be the name of the publisher and address in 16-point Times New Roman.
- The details of the offices of the publisher (only location) should appear in the footer.

5. Create the following one-page documents:

- a) Compose a note inviting friends to a get-together at your house, including a list of things to bring with them.
- b) Design a certificate in landscape orientation with a border around the document.
- c) Design a Garage Sale sign.
- d) Make a sign outlining your rules for your bedroom at home, using a numbered list.

6. 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.

7. Convert following text to a table, using comma as delimiter:

Colour, Style, Item

Blue, A980, Van

Red, X023, Car

Green, YL724, Truck

Name, Age, Sex

Bob, 23, M

Linda, 46, F

Tom, 29, M

8. Enter the following data into a table given on the next page:

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

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.

9. Wrapping of text around the image.

MS Excel

1. Given the following worksheet:

Sl. No.	Roll No.	Name	Marks	Grade
1	1001	Sachin	99	
2	1002	Sehwag	65	
3	1003	Rahul	41	
4	1004	Sourav	89	
5	1005	Harbhajan	56	

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

Marks	Grade
≥ 80	A+
≥ 60 AND < 80	A
≥ 50 AND < 60	B
< 50	F

2. 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

3. Enter the following data in Excel Sheet:

PERSONAL BUDGET FOR FIRST QUARTER

Monthly Income (Net): 1475

EXPENSES	JAN	FEB	MARCH	Quarter Total	Quarter Average
Rent	600.00	600.00	600.00		
Telephone	48.25	43.50	60.00		
Utilities	67.27	110.00	70.00		
Credit Card	200.00	110.00	70.00		
Oil	100.00	100.00	90.00		
AV to Insurance	150.00	150.00	110.00		
Cable TV	40.75	40.75	40.75		

Calculate Quarter Total and Quarter Average:

- Calculate Monthly Total.
- Surplus = Monthly Income – Monthly Total.
- What would be Total Surplus if Monthly Income is 1500?
- How much does telephone expense for March differ from Quarter Average?
- Create a 3D column graph for Telephone and Utilities.
- Create a pie chart for monthly expenses.

GE2 - Introduction to Programming (using C)

Theory: 60 Lectures

1. Overview of C

[7 Lectures]

A Brief History of C, Form of a C Program, Compilation and Execution of a C Program.

2. Data Types, Variables, Constants, Operators and Basic I/O [10 Lectures]

Basic Data Types, Declaration and Definition of Variables, Variable Initialization, Constants, Operators, Console I/O.

3. Expressions and Statements [8 Lectures]

Operator Precedence, Type Conversion in Expressions, Casts, Selection Statements, Iteration Statements, Jump Statements.

4. Arrays, Strings and Functions [12 Lectures]

One-Dimensional Arrays, Array Initialization, Passing One-Dimensional Arrays to Functions, Two-Dimensional Arrays, Passing Two-Dimensional Arrays to Functions, Character Arrays, Strings, Array of Strings.

General form of a Function, Function Declaration and Definition, Function Arguments (*Call by Value and Call by Reference*), Returning from a Function, Functions Returning Values.

5. Derived Data Types (Structures) [5 Lectures]

Basics of Structures, Accessing Structure Members, Structure Assignments, Array of Structures, Pointers to Structures, Passing Structures to Functions, Nested Structures.

6. Pointers in C [8 Lectures]

Pointer Variables, Pointer Assignments, Pointer/Address Arithmetic.

7. Memory Allocation in C [4 Lectures]

Static and Dynamic Memory Allocation, C's Dynamic Allocation Functions. (*malloc()*, *calloc()*, *free()*).

8. File I/O and Preprocessor Directives [6 Lectures]

Opening and Closing a File, Reading and Writing Text Files, Random Access in Files, Preprocessor Directives, Macros.

Introduction to Programming using C Practical:

1. WAP to print the sum and product of digits of an integer.
2. WAP to reverse a number.
3. WAP to compute the sum of the first n terms of the following series

$$S = 1 + 1/2 + 1/3 + 1/4 + \dots$$

4. 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.
5. Write a function to find whether a given number is prime or not. Use the same to generate the prime numbers less than 100.
6. WAP to compute the factors of a given number.

7. Write a macro that swaps two numbers. WAP to use it.
8. Write a program that swaps two numbers using pointers.
9. Write a program in which a function is passed address of two variables and then alter its contents.
10. WAP 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.
11. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically.
12. WAP to display Fibonacci series.
13. WAP to calculate Factorial of a number.
14. WAP to calculate GCD of two numbers.
15. Create a structure Student containing fields for Roll No, Name, Class, Year and Total Marks. Create 10 students and store them in a file.

GE3 - Introduction to Database Systems

Theory: 60 Lectures

1. Introduction [14 Lectures]

Introduction to Databases, DBMS Architecture, Relational Data Model, Database Independence, Database Administrator, Database Users.

2. Entity-Relationship Modelling [16 Lectures]

Entity Types, Entity Sets, Attributes and Keys, Relationships, Relationship Types, ER Diagrams, Database Design using ER Diagrams.

3. Relational Data Model [16 Lectures]

Relational Model Concepts, Relational Constraints, Primary and Foreign Keys, Relational Algebra and Relational Calculus, Normalization (1NF, 2NF, 3NF).

4. Structured Query Language [14 Lectures]

Data Manipulation Language (DML), Data Definition Language (DDL), SQL Queries, Views.

Reference Books:

1. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education, 2010.
2. R. Ramakrishnan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.

3. A. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGraw Hill, 2010.

4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education, 2013.

Database Management Systems

Practical:

1. Create a database having two tables with the specified fields, to computerize a library system of a Delhi University College:

LibraryBooks (AccessionNumber, Title, Author, Department, PurchaseDate, Price)

IssuedBooks (AccessionNumber, Borrower)

a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.

b) Delete the record of book titled “Database System Concepts”.

c) Change the Department of the book titled “Discrete Maths” to “CS”.

d) List all books that belong to CS department.

e) List all books that belong to the CS department and are written by author “Navathe”.

f) List all books which have a price less than 500 or purchased between 01/01/1999 and 01/01/2004.

2. Create the following tables and enter at least 5 records in each table and answer the queries given below:

EMPLOYEE (Person_Name, Street, City)

WORKS (Person_Name, Company_Name, Salary)

COMPANY (Company_Name, City)

MANAGES (Person_Name, Manager_Name)

a) Identify primary and foreign keys.

b) Alter table Employee, add a column Email of type VARCHAR(20).

c) Find the name of all managers who work for both Samba Bank and NCB Bank.

d) Find the names, street address and cities of residence and salary of all employees who work for Samba Bank and earn more than 10000.

e) Find the names of all employees who live in the same city as the company for which they work.

f) Find the highest salary, lowest salary and average salary paid by each company.

g) Find the sum of salary and number of employees in each company.

h) Find the name of the company that pays highest salary.

3. Create the following tables, enter at least 5 records in each table and answer the queries given below:

Suppliers (SNo, SName, Status, SCity)

Parts (PNo, PName, Colour, Weight, City)

Project (JNo, JName, JCity)

Shipment (SNo, PNo, JNo, Quantity)

- a) Identify primary and foreign keys.
- b) Get supplier numbers for suppliers in Paris with status>20.
- c) Get supplier details for suppliers who supply part P2. Display the supplier list in increasing order of supplier numbers.
- d) Get supplier names who do not supply part P2.
- e) For each shipment get full shipment details, including total shipment weights.
- f) Get all the shipments where the quantity is in the range 300 to 750 inclusive.
- g) Get the names of cities that store more than five red parts.
- h) Get full details of parts supplied by a supplier in London.
- i) Get part numbers for part supplied by a supplier in London to a project in London.
- j) Get the total number of project supplied by a supplier (say, S1).
- k) Get the total quantity of a part (say, P1) supplied by a supplier (say, S1).

GE4 - Electronic Commerce (E-commerce)

Theory: 60 Lectures

1. Introduction

[14 Lectures]

Meaning, Nature, Concepts, Advantages, Disadvantages and Reasons for Transacting Online, Types of E-Commerce, E-commerce Business Models (Introduction, Key Elements of a Business Model and Categorizing Major E-commerce Business Models), The dynamics of World Wide Web and Internet (Meaning, Evolution and Features).

2. Security and Encryption

[16 Lectures]

Need and Concepts, The E-commerce Security Environment (Dimension, Definition and Scope of E-security), Security Threats in the E-commerce Environment (Security Intrusions and Breaches, Attacking Methods like Hacking, Sniffing, Cyber-Vandalism), Technology Solutions (Encryption, Security Channels of Communication, Protecting Networks and Protecting Servers and Clients).

3. E-payment System

[14 Lectures]

Models and Methods of E-payments (Debit Card, Credit Card, Smart Cards, E-money), Digital Signatures (Procedure, Working and Legal Position), Payment Gateways, Online

Banking (Meaning, Concepts, Importance, Electronic Fund Transfer), Risks involved in E-payments.

4. Online Business Transactions

[16 Lectures]

Meaning, Purpose, Advantages and Disadvantages of Transacting Online, E-commerce Applications in various Industries like Banking, Insurance, Payment of Utility Bills, Online Marketing, Online Services (Financial, Travel and Career), Auctions, Online Portal, Online Learning, Publishing and Entertainment, Online Shopping.

Reference Books:

1. G.S.V. Murthy, E-Commerce Concepts, Models, Strategies, Himalaya Publishing House, 2011.
2. Kamlesh K Bajaj and Debjani Nag, E-Commerce, 2005.
3. Gray P. Schneider, Electronic commerce, International Student Edition, 2011.
4. Henry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang, E-Commerce: Fundamentals and Applications, Wiley Student Edition, 2011.