

B.Sc. with Computer Science Syllabus

I Semester, DSC 1A

Object Oriented Programming with C++

Unit I

Algorithm and its characteristics, pseudo code / flow chart, program.

Object Oriented Programming: Introduction, Generation of programming Languages, Programming Paradigms, Features of Object Oriented Programming, Merits and Demerits of Object Oriented Programming Language.

Basics of C++ Programming: Introduction, History, Structure, Writing the First C++ Program, Files used in a C++ Program, Compiling and Executing, Using Comments, Tokens, Characters Set, Keywords, Identifier, Data Types, Variables, Constants, Input and Output, Statements, Operators, Type Conversion and Type Casting,

Decision Control and Looping Statements: Introduction to Decision Control Statements, Conditional Branching Statements, Iterative Statements, Nested Loops, Break Statement, Continue Statement, Goto Statement, Avoiding Usage of Break, Continue, and Goto Statements

Unit II

Functions: Introduction, Need for Functions, Using Functions, Function Declaration or function, Prototype, Function Definition, Function Call, Return Statement, Passing Parameters to the Function, Default Arguments, Return by Reference, Passing Constants as Arguments, Variables Scope, Storage Classes, Inline Functions, Function Overloading, Recursive Functions, Recursion Versus Iteration, Function with Variable Number of Arguments

Arrays: Introduction, Declaration of Arrays, Accessing Elements of the Array, Storing Values in Arrays, Calculating the Length of Array, Operations that can be Performed on Arrays, One Dimensional Arrays for Inter Function Communication, Two Dimensional Arrays, Multi Dimensional Arrays,

Pointers: Defining Pointers, Declaring Pointer Variables, Pointer Expressions and Pointer Arithmetic, Null Pointers, Generic Pointers, Passing Arguments to Function Using Pointer, Pointer and Arrays, Passing Array to Function, Differences Between Array Name and Pointer, Pointer to function, Arrays of Function Pointers, Memory Allocation in C++, Dynamic Memory Allocation,

Unit III

Structure, Union, and Enumerated Data Types: Structure Declaration, Typedef Declaration, Initialization the Structures, Accessing the Members of a Structures, Union, Union Inside Structures, Enumerated Data Types.

B.Sc. with Computer Science Syllabus

Classes and Objects : Specifying a Class, Creating Objects, Accessing Object Members, Nested Member Functions, Making a Member Function Inline, Memory Allocation for Class and Objects, Returning Objects, this Pointer, Constant Parameters and Members, Pointers within a Class, Local Classes, Nested Classes in C++, Empty Classes, Friend Function, Friend Class, Bit-Field in Classes, Pointers and Class Members.

Constructors and Destructors: Constructor, Types of Constructors, Constructor with Default Arguments, Constructor Overloading, Destructors.

Unit IV

Operator overloading and Type Conversions : Scope of Operator Overloading, Syntax for Operator Overloading, Operators that can and cannot be Overloaded, Implementing Operator Overloading, Overloading Unary Operators, Overloading Binary Operators, Overloading Special Operators, Type Conversions.

Inheritance and Run-Time Polymorphism : Defining Derived Classes, Access Specifiers, Type of Inheritance, Single Inheritance, Constructors and Destructors in Derived Classes, Multi level Inheritance, Constructor in Multi Level Inheritance, Multiple Inheritance, Constructor and Destructor in Multiple Inheritance, Ambiguity in Multiple Inheritance, Hierarchical Inheritance, Constructors and Destructors in Hierarchical Inheritance, Hybrid Inheritance, Multi-path Inheritance, Virtual Base Classes, Object Slicing, Pointer to Derived Class, Run time Polymorphism, Virtual Functions, Pure Virtual Functions, Abstract Base Classes, Concept of Vtables, Virtual Constructor and Destructor.

Templates: Introduction, Use of templates, Function templates, Class templates.

Text Book:

1. Reema Thareja “Object Oriented Programming with C++” Oxford university Press, 2015

Recommended Books

1. E. Balagurusamy “Object Oriented Programming with C++” TMH, 6th edition, 2013.
2. Richard Johnson, *An Introduction to Object-Oriented Application Development*, Thomson Learning, 2006
3. B. Stroustrup, *The C++ Programming Language*, Addison Wesley, 2004.
4. Spoken Tutorial on “C++” as E-resource for Learning:- <http://spoken-tutorial.org>

B.Sc. with Computer Science Syllabus

Practical: Object Oriented Programming with C++

NOTE:

- All the concepts of programs from Text Book including exercises must be practice, execute and write down in the practical record book.
- Faculty must take care about UG standard programs it should be minimum 25 – 30.
- In the external lab examination student has to execute at least three programs with compilation and deployment steps are necessary.
- External Viva-voce is compulsory.

Example programs:

- 1) Write a program to test Arithmetic operators.
- 2) Write a program to Swap two numbers.
- 3) Write a program to demonstrate Switch statement.
- 4) Write a program to find roots of a quadratic equation.
- 5) Write a program to check whether the given number is palindrome or not.
- 6) Write a program to convert binary number to decimal number.
- 7) Write a program to print the following format.

1			
2	3		
4	5	6	
7	8	9	10
- 8) Write a program to search an element in a given list.
- 9) Write a program to perform addition of two Matrices.
- 10) Write a program to perform multiplication of two Matrices.
- 11) Write a program to find factorial of a given number using recursion.
- 12) Write a program to demonstrate Pointer arithmetic
- 13) Write a program to demonstrate Call-By-Value, Call-By-Address, Call-By-Reference.

B.Sc. with Computer Science Syllabus

- 14) Write a program to demonstrate Structure data type.
- 15) Write a program to demonstrate Enumerated data type.
- 16) Write a program to demonstrate inline functions.
- 17) Write a program to demonstrate Function Overloading.
- 18) Write a c++ program to demonstrate Class concept.
- 19) Write a c++ program on Constructor overloading.
- 20) Write a c++ program on Destructor.
- 21) Write a c++ program for copy constructor.
- 22) Write a c++ program to demonstrate Friend function.
- 23) Write a c++ program for Unary operator overloading (Friend function/Member function).
- 24) Write a c++ program for Binary operator overloading (Friend function/Member function).
- 25) Write a c++ program for Member Function overloading within a class
- 26) Write a c++ program for Single and Multilevel Inheritance.
- 27) Write a c++ program for Overriding of member functions.
- 28) Write a c++ program to demonstrate constructor calling mechanism in inheritance.
- 29) Write a c++ program for Multiple and Hybrid inheritance.
- 30) Write a c++ program to demonstrate pure virtual function implementation.

B.Sc. with Computer Science Syllabus

II Semester, DSC 1B

Data Structures and File Processing

Unit I

Basic data Structure: Introduction to Data Structures, Types of Data Structures, and Introduction to Algorithms, Pseudocode, and Relationship among data, data structures, and algorithms, Implementation of data structures, Analysis of Algorithms.

Stacks: Concept of Stacks and Queues, Stacks, Stack Abstract Data Type, Representation of Stacks Using Sequential Organization (Arrays), Multiple Stacks, Applications of Stack, Expression Evaluation and Conversion, Polish notation and expression conversion, Processing of Function Calls, Reversing a String with a Stack, Recursion.

Memory Management: Garbage collection algorithms for equal sized blocks, storage allocation for objects with mixed size, buddy systems

Unit II

Recursion: Introduction, Recurrence, Use of Stack in Recursion, Variants of Recursion, Recursive Functions, Iteration versus Recursion.

Queues: Concept of Queues, Queue as Abstract Data Type, Realization of Queues Using Arrays, Circular Queue, Multi-queues, Deque, Priority Queue, Applications of Queues,

Linked Lists: Introduction, Linked List, Linked List Abstract Data Type, Linked List Variants, Doubly Linked List, Circular Linked List, Representation of Sparse Matrix Using Linked List, Linked Stack, Linked Queue, Generalized Linked List, More on Linked Lists.

Unit III

Trees: Introduction, Types of Trees, Binary Tree, Binary Tree Abstract Data Type, Realization of a Binary Tree, Insertion of a Node in Binary Tree, Binary Tree Traversal, Other Tree Operations, Binary Search Tree, Threaded Binary Tree, Applications of Binary Trees.

Searching and Sorting: Searching, Search Techniques, Sorting, Multiway Merge and Polyphase Merge, Comparison of All Sorting Methods, Search Trees: Symbol Table, Optimal Binary Search Tree, AVL Tree (Height-balanced Tree).

B.Sc. with Computer Science Syllabus

Unit IV

Hashing: Introduction, Key Terms and Issues, Hash Functions, Collision Resolution Strategies, Hash Table Overflow, Extendible Hashing, Dictionary, Skip List, Comparison of Hashing and Skip Lists.

Heaps: Basic Concepts, Implementation of Heap, Heap as Abstract Data Type, Heap Applications,

Indexing and Multiway Trees: Introduction, Indexing, Types of Search Trees
Files: Introduction, External Storage Devices, File Organization, Sequential File Organization, Direct Access File Organization, Indexed Sequential File Organization, Linked Organization.

Text books:

1. Varsha H. Patil “ Data structures using C++” Oxford university press, 2012
2. M.T. Goodrich, R. Tamassia and D. Mount, *Data Structures and Algorithms in C++*, John Wiley and Sons, Inc., 2011.

Recommended Books

1. Adam Drozdek “Data structures and algorithm in C++” Second edition, 2001
2. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, *Introduction to Algorithms*, 2nd Ed., Prentice-Hall of India, 2006.
3. Robert L. Kruse and A.J. Ryba, *Data Structures and Program Design in C++*, Prentice Hall, Inc., NJ, 1998.
4. B. Stroustrup, *The C++ Programming Language*, Addison Wesley, 2004
5. D.E. Knuth, *Fundamental Algorithms* (Vol. I), Addison Wesley, 1997

B.Sc. with Computer Science Syllabus

Practical: Data Structures and File Processing

NOTE:

- All the concepts of programs from Text Book including exercises must be practice, execute and write down in the practical record book.
- Faculty must take care about UG standard programs it should be minimum 25 – 30.
- In the external lab examination student has to execute at least three programs with compilation and deployment steps are necessary.
- External Viva-voce is compulsory.

Example programs:

1. Write C++ programs to implement the following using an array
 - a) Stack ADT
 - b) Queue ADT
2. Write a C++ program to implement Circular queue using array.
3. Write C++ programs to implement the following using a single linked list.
 - a) Stack ADT
 - b) Queue ADT
4. Write a C++ program to implement Circular queue using Single linked list.
5. Write a C++ program to implement the double ended queue ADT using double linked list.
6. Write a C++ program to solve tower of hanoi problem recursively
7. Write C++ program to perform the following operations:
 - a) Insert an element into a binary search tree.
 - b) Delete an element from binary search tree.
 - c) Search for a key in a binary search tree.
8. Write C++ programs for the implementation of BFS and DFS.
9. Write a C++ program that uses non-recursive functions to traverse a binary tree.

B.Sc. with Computer Science Syllabus

- a)Pre-order
- b)In-order
- c)Post-order

10. Write a C++ program to find height of a tree.

11 Write a C++ program to find MIN and MAX element of a BST.

12 Write a C++ program to find Inorder Successor of a given node.

13. Write C++ programs to perform the following operations on B-Trees and AVL Trees.

- a)Insertion
- b)Deletion

14 Write C++ programs for sorting a given list of elements in ascending order using the following sorting methods.

- a)Quick sort
- b)Merge sort

15. Write a C++ program to find optimal ordering of matrix multiplication.

16. Write a C++ program that uses dynamic programming algorithm to solve the optimal binary search tree problem

17. Write a C++ program to implement Hash Table

18. Write C++ programs to perform the following on Heap

- a)Build Heap
- b)Insertion
- c)Deletion

19. Write C++ programs to perform following operations on Skip List

- a)Insertion
- b)Deletion

20. Write a C++ program to Heap sort using tree structure.

SECOND YEAR
THEORY PAPER - II
OBJECT ORIENTED PROGRAMMING WITH JAVA
AND DATA STRUCTURES.

UNIT - 1: Java Fundamentals

Fundamentals of Object Oriented programming : Object Oriented paradigm - Basic concepts of Object Oriented Programming - Benefits of OOP - Applications of OOP.

Java Evolution : Java Features - How Java differs from C and C++ - Java and Internet - Java and World Wide Web - Web Browsers - Hardware and Software Requirements - Java Environment. Overview of Java Language: Simple Java Program - Java Program Structure - Java Tokens- Java Statements - Implementing a Java Program - Java Virtual Machine - Command Line Arguments. Constants, Variables and Data types: Constants - Variables - Data types - Declaration of Variables- Giving Values to variables- Scope of Variables-Symbolic Constants-Type Casting. (Chapters: 1,2,3,4)

UNIT - 2: Oops Concepts in Java

Operators and Expressions: Arithmetic Operators - Relational Operators- Logical Operators - Assignment Operators - Increment and Decrement Operators - Conditional Operators - Bitwise Operators - Special Operators - Arithmetic Expressions - Evaluation of Expressions - Precedence of Arithmetic Operators - Operator Precedence and Associativity.

Decision Making and Branching: Decision Making with If statement - Simple If Statement-If else Statement-Nesting If else Statement-the elseif Ladder-The switch Statement - The ?: operator.

Decision Making and Looping: The while statement - The do statement - The for statement - Jumps in Loops. Class , Objects and Methods: Defining a Class - Fields Declaration - Methods Declaration - Creating.

Objects - Accessing class members - Constructors - Methods Overloading - Static Members - Nesting of Methods - Inheritance - Overriding Methods - Final Variables and Methods - Final Classes - Abstract Methods and Classes - Visibility Control. (Chapters : 5,6,7,8)

UNIT - 3: Packages and Interfaces in Java

Arrays, Strings and Vectors: One-dimensional Arrays-creating an Array - Two dimensional Arrays - Strings - Vectors - Wrapper Classes - Enumerated Types.

Interfaces: Multiple Inheritance : Defining Interfaces - Extending Interfaces - Implementing Interfaces - Accessing Interface Variables.

Packages: Java API Packages - Using system Packages - Naming Conventions - Creating Packages - Accessing a Package - Using a Package - Adding a Class to a Package - Hiding Classes - Static Import. (CHAPTERS: 9,10,11)

UNIT - 4: Multithreaded programming and Applets.

Multithreaded Programming: Creating Threads - Extending the Thread Class - Stopping and Blocking a Thread - Life Cycle of a Thread - Using Thread Methods - Thread Exceptions - Thread Priority -Synchronization. Managing Errors and Exceptions: Types of Errors - Exceptions - Syntax of Exception Handling Code - Multiple Catch Statements - Using Finally Statement - Throwing our own Exceptions - Using Exceptions for debugging.

Applet Programming: How Applets differ from Applications - Preparing to write Applets - Building Applet Code - Applet Life Cycle - Creating an executable Applet - Designing a WebPage - Applet Tag - Adding Applet to HTML file - Running the Applet - More about Applet Tag - Passing parameters to Applets - Aligning the display - More about HTML tags - Displaying Numerical Values - Getting Input from the user. (Chapters: 12, 13, 14)

UNIT - 5: Data Structures

Sorting: Bubble Sort - Selection Sort - Insertion Sort - Quick Sort- Stacks and Queues: Stacks - Queues - Circular Queue - Deques - Priority Queue - Parsing Arithmetic Expressions - Linked List: Simple Linked List - Finding and Deleting Specified Links - Double Ended Lists - Abstract Data types - Sorted Lists - Doubly Linked Lists - Advanced Sorting : Quick Sort - Binary Trees : Tree Terminology - Finding a Node - Inserting a Node - Traversing the Tree - Finding Maximum and Minimum values - Deleting a Node - Efficiency of Binary Trees - Trees Represented as Arrays - Graphs: Introduction to Graphs - Searches - Minimum Spanning Tree - Topological Sorting with Directed Graphs - Connectivity in Directed Graphs. (Chapters : 3,4,5,7 (Only Quick Sort), 8,13)

Prescribed books:

1. E.Balaguruswamy, Programming with Java, A primer, 3e, TATA McGraw-Hill Company (2008).(Chapters : 1 to 14)
2. Robert Lafore, Data Structures & Algorithms in Java, Second Edition, Pearson Education(2008) (Chapters: 3,4,5,7 (Only Quick Sort),8,13)

Reference Books :

1. John R. Hubbard, Programming with Java, Second Edition, Schaum's outline Series, Tata McGrawhill (2007).
2. Timothy Budd, Understanding Object Oriented Programming with Java, Pearson Education (2007).
3. Adam Drozdek, Data Structures and Algorithms in Java, Second Edition, Cengage Learning(2008).
4. John R. Hubbard, Anita Hurry, Data Structures with Java, Pearson Education (2008).
5. Jana, Java and Object Oriented Programming Paradigm, PHI (2007).
6. Deitel & Deitel. Java TM: How to Program, 7th Edition, PHI (2008).
7. Samatha, Classic Data Structures, PHI (2005).

SECOND YEAR**PRACTICAL PAPER - II****JAVA AND DATA STRUCTURES LAB****Java Lab Cycle**

1. Write a Java program to determine the sum of the following harmonic series for a given value of 'n'. $1 + 1/2 + 1/3 + \dots + 1/n$
2. Write a program to perform the following operations on strings through interactive input.
 - a) Sort given strings in alphabetical order.
 - b) Check whether one string is sub string of another string or not.
 - c) Convert the strings to uppercase.
3. Write a program to simulate on-line shopping.
4. Write a program to identify a duplicate value in a vector.
5. Create two threads such that one of the thread print even no's and another prints odd no's up to a given range.

6. Define an exception called “Marks Out Of Bound” Exception, that is thrown if the entered marks are greater than 100.
7. Write a JAVA program to shuffle the list elements using all the possible permutations.
8. Create a package called “Arithmetic” that contains methods to deal with all arithmetic operations. Also,” write a program to use the package.
9. Write an Applet program to design a simple calculator.
10. Write a program to read a text and count all the occurrences of a given word. Also, display their positions.
11. Write an applet illustrating sequence of events in an applet.
12. Illustrate the method overriding in JAVA.
13. Write a piogram to fill elements into a list. Also, copy them in reverse order into another list.
14. Write an interactive program to accept name of a person and validate it. If the name contains any numeric value throw an exception “InvalidName”.
15. Write an applet program to insert the text at the specified position.
16. Prompt for the cost price and selling price of an article and display the profit (or) loss percentage.
17. Create an anonymous array in JAVA.
18. Create a font animation application that changes the colors of text as and when prompted.
19. Write an interactive program to wish the user at different hours of the day.
20. Simulate the library information system i.e. maintain the list of books and borrower’s details.

Data Structures Lab Cycle

21. Program to create, insert, delete and display operations on single linked list?
22. Program to create , insert, delete and display operations on double linked list ?
23. Program to create , insert, delete and display operations on circular single linked list ?

24. Program to split a single linked list
25. Program to reverse a single linked list
26. Program to implement Insertion Sort.
27. Program to implement PUSH and POP operations on Stack using array method.
28. Program to implement PUSH and POP operations on Stack using Linked list method.
29. Program to implement insert and delete operations on Queue using array method.
30. Program to implement insert and delete operations on Queue using linked list method.
31. Program to implement insert and delete operations on Priority Queue?
32. Program to implement insert and delete operations on Double Ended Queue?
33. Program to evaluate postfix expression by using Stack?
34. Program to construct Binary Search Tree and implement tree traversing Techniques.
35. Program to delete a leaf node from binary search tree.
36. Program to implement Selection Sort.
37. Program to implement Bubble Sort.
38. Program to implement Operations on Circular Queue.
39. Program to implement Quick Sort.
40. Program to Find number of Leaf nodes and Non-Leaf nodes in a Binary Search Tree.
41. Program for Insertion Sort.

THIRD YEAR

THEORY PAPER - III : DATABASE MANAGEMENT SYSTEMS

UNIT-1

Database Systems: Introducing the database and DBMS, Why the database is important, Historical Roots: Files and File Systems, Problems with File System Data Management, Database Systems.

Data Models: The importance of Data models, Data Model Basic Building Blocks, Business Rules, The evaluation of Data Models, Degree of Data Abstraction.

The Relational Database Model: A logical view of Data, Keys, Integrity Rules, Relational Set Operators, The Data Dictionary and the system catalog, Relationships within the Relational Database, Data Redundancy revisited, Indexes, Codd's relational database rules. (Chapters -1: 1.2 to 1.6,2,3)

UNIT-2

Entity Relationship Model: The ER Model, Developing ER Diagram, Database Design Challenges: Conflicting Goals.

Advanced Data Modeling: The Extended Entity Relationship Model, Entity clustering, Entity integrity: Selecting Primary keys, Design Cases: Learning Flexible Database Design.

Normalization of database tables: Database Tables and Normalization, The need for Normalization, The Normalization Process, Improving the design, Surrogate Key Considerations, High level Normal Forms, Normalization and database design, denormalization. (Chapters: 4,6,5)

UNIT - 3

Introduction to SQL: Data Definition Commands, Data Manipulation Commands, Select queries, Advanced Data Definition Commands, Advanced Select queries, Virtual Tables, Joining Database Tables.

Advanced SQL: Relational Set Operators, SQL Join Operators, Subqueries and correlated queries, SQL Functions, Oracle Sequences, Updatable Views, and Procedural SQL.

Database Design: The Information System, The Systems Development Life Cycle, The Database Life Cycle, Database Design Strategies, Centralized Vs Decentralized design. (Chapters: 7,8(8.1 to 8.7),9)

UNIT - 4

Transaction Management and Concurrency Control: What is transaction, Concurrency control, Concurrency control with locking Methods, Concurrency control with time stamping methods, concurrency control with optimistic methods, database recovery management.

Distributed Database Management Systems: The evolution of Distributed Database Management Systems, DDBMS advantages and Disadvantages, Distribution Processing and Distribution Databases, Characteristics of Distributed database management systems, DDBMS Components, Levels of Data and Process distribution, Distributed database Transparency Features, Distributed Transparency, Transaction Transparency, Performance Transparency and Query Optimization, Distributed Database Design, Client Server Vs DDBMS. (Chapters: 10, 12)

UNIT - 5

The Data Warehouse: The need for data analysis, Decision support systems, The data warehouse, Online Analytical Processing, Star schemas, Data mining, SQL extension for OLAP.

Database Administration: Data as a Corporate asset, The need for and role of databases in an organization, The evolution of the database administration function, The database environment's Human Component, Database administration Tools, The DBA at work: Using Oracle for Database Administration. (Chapter: 13:13.1 to 3.5,13.7,13.8,15:15.1,15.2,15.4,15.5,15.6,15.8)

Prescribed Text Book:

1. Peter Rob, Carlos Coronel, Database Systems Design, Implementation and Management, Seventh Edition, Thomson (2007)

Reference Books:

1. Elimasri / Navathe, Fundamentals of Database Systems, Fifth Edition, Pearson Addison Wesley (2007).

2. Raman A Mata - Toledo/Panline K Cushman, Database Management Systems, Schaum's Outlibe series, Tata McGraw Hill (2007).
3. C.J.Date, A.Kannan, S.Swamynathan, An Introduction to Database Systems, Eight Edition, Pearson Education (2006).
4. Michel Kifer, Arthur Bernstein, Philip M. Lewis, Prabin K. Pani Graphi, Database Systems: An application oriented Approach, seond edition, pearson education (2008).
5. Atul Kahate, Introduction to Database Management Systems, Pearson Education (2006).

PAPER - 4.2 :

GUI PROGRAMMING

Detailed Syllabus

UNIT-1

Getting Started with Visual Basic 6.0: Introduction to Visual Basic, Visual Basic 6.0 Programming Environment, working with Forms, Developing an Application, Variables, Data types and Modules, Procedures and Control Structures, Arrays in Visual Basic

Working with Controls: Introduction, Creating and Using Controls, Working with Control Arrays.

Menus, Mouse Events and Dialog Boxes: Introduction, Mouse Events, Dialog Boxes. (Chapters:1,2,3)

UNIT - 2

Graphics, MDI and Felx Grid: Introduction, Graphics for application, Multiple Document Interface(MDI), Using FlexGrid Control.

Object Linking and Embedding: Introduction, OLE Fundamentals, Using OLE Container Control, Using OLE Automation Objects, OLE Drag and Drop.

Objects and Classes: Introduction to Objects. Working with Objects, Classes and Class Modules.

Working with Add-Ins: Introduction to Add-Ins, Building Add-Ins. (Chapters: 4, 8, 9, 14)

UNIT - 3

File and File system Controls: Introduction, File System Controls, Accessing Files, Interface with Windows.

ODBC and Data Access Objects: Evolution of Computing Architectures, Data Access Options.

ODBC using Data Access Objects and Remote Data Objects: Open Database Connectivity, Remote Data Objects.

Working with ActiveX Data Objects: An overview of ADO and OLEDB, ADO object Model. (Chapters: 17,5,6,16)

UNIT - 4

Data Environment and Data Report: Introduction, Data Environment Designer, Data Report.

All about ActiveX Controls: Introduction, Constituents of ActiveX Control, Exposing ActiveX Control Properties.

ActiveX EXE and ActiveX DLL: Introduction to ActiveX EXE and ActiveX DLL, Creating and ActiveX EXE Component, Creating an ActiveX DLL Component. (Chapters: 7,10,11)

UNIT - 5

ActiveX Document Fundamentals: What is an ActiveX Document, Active Server Pages.

Built-in ActiveX Controls: Working with Built-in ActiveX Controls, Additional ActiveX Controls.

Introducing Web Browser and DHTML: Introduction, Internet Tools in Visual Basic, Using DHTML in Visual Basic. (Chapters: 12,13,15)

Prescribed Text Book:

1. Content Development Group, Visual Basic 6.0 Programming, Tata McGraw-Hill Publishing Company Limited (2007).

Reference Books :

1. Deitel and Deitel, Visual Basic 2005, Third Edition, Pearson Education (2007).
2. Noel Jerke, Visual Basic 6, The complete reference, Tata McGraw Hill (2006).
3. Byran S. Gottfried, Visual Basic, Schaum's outlines, Tata McGraw Hill (2004).

B.Sc. with Computer Science Syllabus

III Semester, DSC 1C

Database Management System

Unit I

Introduction: Database-System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Database Users and Administrators.

Introduction to the Relational Model: Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relational Query Languages, Relational Operations.

Unit II

Database Design and the E-R Model: Overview of the Design Process, The Entity-Relationship Model, Constraints, Removing Redundant Attributes in Entity Sets, Entity-Relationship Diagrams, Reduction to Relational Schemas, Entity-Relationship Design Issues, Extended E-R Features, Alternative Notations for Modeling Data, Other Aspects of Database Design.

Relational Database Design: Features of Good Relational Designs, Atomic Domains and First Normal Form, Decomposition Using Functional Dependencies, Functional-Dependency Theory, Decomposition Using Multivalued Dependencies, More Normal Forms, Database-Design Process.

Unit III

Database-System Architectures: Centralized and Client –Server Architectures, Server System Architectures, Parallel Systems, Distributed Systems, Network Types.

Introduction to SQL: Overview of the SQL Query Language, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Subqueries, Modification of the Database.

Unit IV

Intermediate SQL: Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization.

B.Sc. with Computer Science Syllabus

Advanced SQL: Accessing SQL From a Programming Language, Functions and Procedures, Triggers, Recursive Queries.

Text book:

1. A. Silberschatz, H. Korth and S. Sudarshan, *Database System Concepts*, 6th Ed., Tata McGraw Hill, 2011

References:

1. J. Morrison, M. Morrison and R. Conrad, *Guide to Oracle 10g*, Thomson Learning, 2005.
2. Loney and Koch, *Oracle 10g: The Complete Reference*, Tata McGraw Hill, 2006.
3. David Flanagan, Java Script, *The Definitive Guide*, O'Reilly Media, 2006.
4. Marty Hall, Larry Brown, and Yaakov Chaikin, *Core Servlets and Java Server Pages: Core Technologies* (Vol. II), 2nd Ed., Sun Microsystems Press, 2006.
5. S.K. Singh, *Database Systems Concepts, Design and Applications*, Pearson Education 2006.
6. Spoken Tutorial on "MySQL" as E-resource for Learning:- <http://spoken-tutorial.org>

B.Sc. with Computer Science Syllabus

Practical: Database Management System

NOTE:

- All the concepts of programs from Text Book including exercises must be practice, execute and write down in the practical record book.
- Faculty must take care about UG standard programs it should be minimum 25 – 30.
- In the external lab examination student has to execute at least three programs with compilation and deployment steps are necessary.
- External Viva-voce is compulsory.

Example programs:

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

LibraryBooks (Accession number, Title, Author, Department, PurchaseDate, Price)

IssuedBooks (Accession number, 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 “CS” department and are written by author “Navathe”.
 - f) List all computer (Department=“CS”) that have been issued.
 - g) List all books which have a price less than 500 or purchased between “01/01/1999” and “01/01/2004”.
2. Create a database having three tables to store the details of students of Computer Department in your college.

Personal information about Student (College roll number, Name of student, Date of birth, Address, Marks(rounded off to whole number) in percentage at 10 + 2, Phone number)

Paper Details (Paper code, Name of the Paper)

Student’s Academic and Attendance details (College roll number, Paper code, Attendance, Marks in home examination).

- a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.
 - b) Design a query that will return the records (from the second table) along with the name of student from the first table, related to students who have more than 75% attendance and more than 60% marks in paper 2.
 - c) List all students who live in “Delhi” and have marks greater than 60 in paper 1.
 - d) Find the total attendance and total marks obtained by each student.
 - e) List the name of student who has got the highest marks in paper 2.
3. Create the following tables and answer the queries given below:

B.Sc. with Computer Science Syllabus

Customer (CustID, email, Name, Phone, ReferrerID)

Bicycle (BicycleID, DatePurchased, Color, CustID, ModelNo)

BicycleModel (ModelNo, Manufacturer, Style)

Service (StartDate, BicycleID, EndDate)

- a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.
 - b) List all the customers who have the bicycles manufactured by manufacturer "Honda".
 - c) List the bicycles purchased by the customers who have been referred by customer "C1".
 - d) List the manufacturer of red colored bicycles.
 - e) List the models of the bicycles given for service.
4. Create the following tables, 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 \$10,000.
 - 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.
5. 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, Qunatity)

- a) Identify primary and foreign keys.
- b) Get supplier numbers for suppliers in Paris with status>20.
- c) Get suppliers details for suppliers who supply part P2. Display the supplier list in increasing order of supplier numbers.
- d) Get suppliers names for suppliers 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.

B.Sc. with Computer Science Syllabus

- g) Get part nos. for parts that either weigh more than 16 pounds or are supplied by suppliers S2, or both.
- h) Get the names of cities that store more than five red parts.
- i) Get full details of parts supplied by a supplier in Delhi.
- j) Get part numbers for part supplied by a supplier in Allahabad to a project in Chennai.
- k) Get the total number of project supplied by a supplier (say, S1).
- l) Get the total quantity of a part (say, P1) supplied by a supplier (say, S1).

B.Sc. with Computer Science Syllabus

IV Semester, DSC 1D

Design and Analysis of Algorithms

Unit I

Fundamentals of the Analysis of Algorithm Efficiency: The Analysis Framework, Asymptotic Notations and Basic Efficiency Classes.

Divide-and-Conquer: maximum-subarray problem, Strassen's algorithm for matrix multiplication, The substitution method for solving recurrences, The recursion-tree method for solving recurrences, The master method for solving recurrences.

Dynamic Programming: Rod cutting, Matrix-chain multiplication, Elements of dynamic programming, longest common subsequence, Optimal binary search trees.

Greedy Algorithms: An activity-selection problem, Elements of the greedy strategy, Huffman codes, Matroids and greedy methods, task-scheduling problem as a matroid.

Unit II

Searching and Sorting Techniques: Review of elementary sorting techniques-selection sort, Bubble sort, insertion sort, more sorting techniques-quick sort, heap sort, merge sort, shell sort, external sorting.

Limitations of Algorithm: Lower-Bound Arguments, Decision Trees, P , NP , and NP -Complete Problems.

Polynomials and the FFT: Representing polynomials, The DFT and FFT, Efficient FFT implementations.

Number-Theoretic Algorithms: Elementary number-theoretic notions, Greatest common divisor(GCD), Modular arithmetic, Addition and Multiplication of two large numbers.

Unit III

String Matching: The naive string-matching algorithm, The Rabin-Karp algorithm, String matching with finite automata, The Knuth-Morris-Pratt algorithm.

B.Sc. with Computer Science Syllabus

NP-Completeness: Polynomial time, Polynomial-time verification, NP-completeness and reducibility, NP-completeness proofs, NP-complete problems.

Approximation Algorithms: The vertex-cover problem, The traveling-salesman problem, The set-covering problem, Randomization and linear programming, The subset-sum problem.

Unit IV

Elementary Graph Algorithms: Representations of graphs, Breadth-first search, Depth-first search, Topological sort, strongly connected components.

Minimum Spanning Trees: Growing a minimum spanning tree, the algorithms of Kruskal and Prim.

Single-Source Shortest Paths: The Bellman-Ford algorithm, Single-source shortest paths in directed acyclic graphs, Dijkstra's algorithm, Difference constraints and shortest paths, Proofs of shortest-paths properties.

Text book:

1. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, *Introduction to Algorithms*, MIT press, 3rd edition, 2009.
2. Anany Levitin, *Introduction to the design and analysis of algorithms*, 3rd edition, 2012.

References:

1. J. Kleinberg and E. Tardos, *Algorithms Design*, Pearson Education, 2006.
2. S. Baase, *Computer Algorithms: Introduction to Design and Analysis*, Addison Wesley, 1999.
3. A.V. Levitin, *Introduction to the Design and Analysis of Algorithms*, Pearson Education, 2006.

B.Sc. with Computer Science Syllabus

Practicals: Design and Analysis of Algorithms

NOTE:

- All the concepts of programs from Text Book including exercises must be practice, execute and write down in the practical record book.
- Faculty must take care about UG standard programs it should be minimum 25 – 30.
- In the external lab examination student has to execute at least three programs with compilation and deployment steps are necessary.
- External Viva-voce is compulsory.

Example programs:

Sorting Algorithm:

1. To analyze time complexity of insertion sort
2. To analyze time complexity of Quick sort
3. To analyze time complexity of merge sort

Dynamic Algorithm:

4. To implement largest common subsequence
5. To implement optimal binary search tree
6. To implement martrix chain multiplication

Divide And Conquer:

7. Implement Binary Search Algorithm.
8. Implement Merge Sort Algorithm.
9. Implement Quick Sort Algorithm.
10. To implement strassen's martrix multiplication algorithm

The Greedy Method:

11. Implement activity selection problem
12. Implement fractional Knapsack Problem Algorithm.
13. Implement Job Sequencing with Deadlines Algorithm.

B.Sc. with Computer Science Syllabus

14. Implement Minimum-Cost Spanning Trees: Prim's Algorithm.
15. Implement Single Source Shortest Paths: Dijkstra's Algorithm.

Dynamic Programming:

16. Implement Single-Source Shortest Paths: Bellman-Ford's Algorithm.
17. Implement All-Pairs Shortest Paths: Floyd & Warshall's Algorithm.

Graphs:

18. Implement Dijkstra's algorithm
19. Implement Warshall algorithm
20. Implement Bellman Fords algorithm
21. Implement depth first search algorithm
22. Implement depth first search algorithm

String Matching Algorithm:

23. Implement Naïve string matching algorithm
24. Implement Rabin Karp string matching algorithm

Spanning Trees:

25. Implement prim's algorithm
26. Implement Kruskal's algorithm

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year
SEMESTER - V:

Paper- : Programming in Java

Unit I

Introduction: Java Essentials, JVM, Java Features, Creation and Execution of Programs, Data Types, Type Conversion, Casting, Conditional Statements, Loops, Branching Mechanism, Classes, Objects, Class Declaration, Creating Objects, Method Declaration and Invocation, Method Overloading,

Unit II

Constructors – Parameterized Constructors, Constructor Overloading, Cleaning-up unused Objects. **Class Variables & Method-static Keyword**, this Keyword, One-Dimensional Arrays, Two-Dimensional Arrays, Command-Line Arguments, Inner Class.

Inheritance: Introduction, Types of Inheritance, extends Keyword, Examples, Method Overriding, super, final Keyword, Abstract classes, Interfaces, Abstract Classes Verses Interfaces.

Packages: Creating and Using Packages, Access Protection, Wrapper Classes, String Class, String Buffer Class.

Unit III

Exception: Introduction, Types, Exception Handling Techniques, User-Defined Exception.

Multithreading: Introduction, Main Thread and Creation of New Threads –By Inheriting the Thread Class or Implementing the Runnable Interface, Thread Lifecycle, Thread Priority and Synchronization.

Input/Output: Introduction, java.io Package, File Class, FileInputStream Class, FileOutputStream Class, Scanner Class, BufferedInputStream Class, BufferedOutputStream Class, RandomAccessFile Class.

Unit IV

Applets: Introduction, Example, Life Cycle, Applet Class, Common Methods Used in Displaying the Output.

Event Handling: Introduction, Types of Events, Example.

AWT: Introduction, Components, Containers, Button, Label, Checkbox, Radio Buttons, Container Class, Layouts. **Swing:** Introduction, Differences between Swing and AWT, JFrame, Japplet, JPanel, Components in Swings, Layout Managers, Jtable, Dialog Box.

Database Handling Using JDBC: Introduction, Types of JDBC Drivers, Load the Driver, Establish Connection, Create Statement, Execute Query, Iterate Resultset, Scrollable Resultset, Developing a JDBC Application.

Text Book:

Sachin Malhotra, Saurabh Choudhary, Programming in Java (2e)

References:

1. Bruce Eckel, Thinking in Java (4e)
2. Herbert Schildt, Java: The Complete Reference (9e)
3. Y. Daniel Liang, Introduction to Java Programming (10e)
4. Paul Deitel, Harvey Deitel, Java: How To Program (10e)
5. Cay S. Horstmann, Core Java Volume I –Fundamentals (10e)
6. C. Thomas Wu, An introduction to object-oriented programming with Java (5e)
7. Tony Gaddis, Starting Out with Java From Control Structures Through Objects (6e)
8. Jeanne Boyarsky, Scott Selikoff, OCA: Oracle Certified Associate Java SE 8 Programmer–I Study Guide

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year
SEMESTER - V:

Programming in Java Lab

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
- Faculty must take care about UG Standard Programs.
- In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
- External Vice-Voce is compulsory.

1. Write Java programs to find the following
 - a) largest of given three numbers
 - b) reverses the digits of a number
 - c) given number is prime or not
 - d) GCD of given two integers
2. Write Java programs to implement the following
 - a) default constructor b) parameterized constructor c) constructor overloading
3.
 - a) Write a Java program to find the smallest from given list of integers using array and scanner class.
 - b) Write a Java program for multiplication of two matrices.
4.
 - a) Write a Java program for demonstrating an inner class or nested class.
 - b) Write a Java program to implement method overloading, method overriding, dynamic method dispatch
5. Write a Java program to implement single, multilevel, hierarchal, multiple, hybrid inheritances.
6. Write Java programs that demonstrate the use of abstract, this, super, static, final keywords
7.
 - a) Write a Java program for creating a package and using a package.
 - b) Write a Java program to demonstrate the use of wrapper classes.
8.
 - a) Write a Java program using all five keywords of exception handling mechanism.

- b) Write a Java program for creating customized (user) Exception
- 9.
- a) Write a Java program that checks whether a given string is a palindrome or not.
 - b) Write a Java program for sorting a given list of names in ascending order.
- 10.
- a) Write a Java program to create a file, write the data and display the data.
 - b) Write a Java program that reads a file name from user and displays its information.
- 11.
- a) Write a Java program for controlling main thread.
 - b) Write a Java program for creating new thread by extending Thread class.
- 12.
- a) Write a Java program for creating new thread by implementing Runnable interface.
 - b) Write a Java program for thread synchronization.
- 13.
- a) Write a Java program to create following AWT components: Button, Checkbox, Choice and List.
 - b) Write Java programs to create AWT application using containers and layouts.
- 14.
- a) Write Java programs to create a simple Applet.
 - b) Write a Java program to handle different types of events in a swing application.
15. Write Java programs to create a swing application using swing components and layouts.
16. Write a Java program to store and retrieve data from database using JDBC.

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year
SEMESTER - V:

Elective 1

A) Computer Networks

Unit I

Introduction: Data Communication Components, Line Configuration, Topologies, Transmission Mode, Categories of Networks, ISO Reference Model–Layered Architecture, Functions of Layers, TCP/IP Reference Model.

Transmission Media: Guided Media–Twisted Pair Cable, Coaxial Cable, Optical Fiber, Unguided Media– Satellite Communication, and Cellular Telephony. Multiplexing: Frequency–Division Multiplexing, Time–Division Multiplexing.

Unit II

Data Link Layer: Error Detection–VRC, LRC, CRC, Checksum, Error Correction–Hamming Code, Burst Error Correction, Line Discipline–ENQ/ACK, Poll/Select, Flow Control–Stop-and-Wait, Sliding Window, Error Control–Stop-and-Wait ARQ, Sliding Window ARQ Go-Back-n ARQ, Selective-Reject ARQ.

Unit III

Local Area Networks: Introduction to IEEE 802, Ethernet-CSMA/CD, Implementation, Token Ring, -Token Passing, Implementation.

Switching: Circuit Switching, Packet Switching, Message Switching.

Unit IV

Networking and Internetworking Devices: Repeaters, Bridges, Routers, Gateways, Routers, Switches, Distance Vector Routing Algorithm, Link State Routing Algorithm.

Transport Layer: Duties of Transport Layer, Connection. Upper OSI Layers; Session Layer, Presentation Layer, Application Layer.

Text Book:

Behrouz A. Forouzan, Data Communication and Networking (2e Update)

References:

1. S.S. Shinde, Computer Networks
2. William Stallings, Data and Computer Communications
3. Andrew S. Tanenbaum, David J Wetherall, Computer Networks
4. Behrouz A Forouzan, Firouz Mosharrarf, Computer Networks A Top-Down Approach
5. James F. Kurose, Keith W. Ross, Computer Networking: A Top-Down Approach Featuring the Internet.

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year
SEMESTER - V:

Elective 1

A) Computer Networks Lab

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
- Faculty must take care about UG Standard Programs.
- In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
- External Vice-Voce is compulsory.

1. Write a program to create a socket and implement connect function.
2. Write a program to get MAC address.
3. Write a program to display hello world using signals.
4. Write a program for socket pair system call using IPC.
5. Write a program to implement the sliding window protocol.
6. Write a program to identify the category of IP address for a given IP address.
7. Write a program to print details of DNS host.
8. Write a program to implement listener and talker.
9. Write a program to implement TCP echo using client–server program.
10. Write a program to implement UDP echo using client–server program.
11. Write a UDP client–server program to convert lowercase letters to uppercase letters.
12. Write a TCP client–server program to convert a given string into reverse.
13. Write a UDP client–server program to convert a given string into reverse.
14. Write a program to implement TCP iterative client–server program.
15. Write a program to implement time service using TCP client–server program.
16. Write a program to implement time service using UDP client–server program.

Note: Write above program using ‘C’ or C++

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year
SEMESTER - V:

Elective 1

B. Visual Programming

Unit I

Introduction to VB: Writing windows application with VB, Programming languages -procedural, object oriented, event driven; VB Environment, Writing first VB project, compiling, debugging, and running the programs.

Controls : Introduction to controls textboxes, frames, check boxes, option buttons, images, setting borders and styles, the shape control, the line control, working with multiple controls and their properties, designing the user interface, keyboard access, tab controls, default & cancel property, coding for controls.

Variables, constants, and Calculation: Data types, naming rules and conversion, constants-named and intrinsic, declaring variables, scope of variables, val function, arithmetic operations, formatting data Counting and accumulating Sums.

Unit II

Decisions and Conditions : If statement, Conditions comparing numeric variables and constants, comparing strings, compound conditions (and, or, not), nested if statements, using if statements with option buttons & check boxes, displaying message in message box, input validation. Calling event procedures, debugging VB projects, Debugging Step-by-Step Tutorial.

Modular programming: Menus, using common dialog box, writing general procedure.

Unit III

Forms Handling: Multiple forms, creating, adding, removing forms, hide, show method, load, unload statement, me keyword, referring to objects on a different forms, Variables and constants in Multiple-Forms.

Iteration Handling: Lists Boxes and Combo Boxes, Do/loops, for/next loops, using msgbox function, using string function

Arrays: control Arrays, the case structure, single-dimension arrays, for Each/Next statement, table lookup, using list boxes with array, multi dimensional arrays.

Unit IV

Database Connectivity: VB and database, using the data control, viewing a database file-step-by-step, Navigating the Database in code, using list boxes and comboboxes as data-bound controls, adding a lookup table and navigation-step-by-step, updating a database file, Record sets, working with database fields, creating a new Dynaset.

Advanced topics in VB: ActiveX controls, Dynamic link libraries (DLL), Multiple Document interface (MDI).

Text Book:

1. Programming in Visual Basic 6.0 by Julia Case Bradley, Anita C. Millispangh (Tata McGraw Hill Edition 2000 (Fourteenth Reprint 2004))

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year
SEMESTER - V:

Elective 1

B) Visual Programming Lab

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
 - Faculty must take care about UG Standard Programs.
 - In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
 - External Vice-Voce is compulsory.
-
1. Print a table of numbers from 5 to 15 and their squares & Cubes.
 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. Complete 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.
 7. Read n numbers. Count the number of negative numbers, positive numbers and zeroes in the list(use arrays)
 8. Read a single dimension array. Find the sum and average of these numbers.
 9. Read a two dimension array. Find the sum of two 2D Array
 10. Write a program to Demonstrate Control Array.
 11. Write a Program to perform String Manipulation Operations.
 12. Develop a VB Application to check for Input Validations.
 13. Develop a VB Application to Demonstrate MDI.
 14. Develop a VB Application to Demonstrate Combobox and Listbox.
 15. Develop a VB Application to Demonstrate Option Buttons and Check Boxes.
 16. Develop a VB Application to deal the following Database Operations
 - a) Insert
 - b) Delete
 - c) Update
 - d) Display

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year
SEMESTER - V:

Elective 1

C) Python

Unit I

Introduction to Python: Python, Features of Python, Execution of a Python Program, Viewing the Byte Code, Flavors of Python, Python Virtual Machine, Frozen Binaries, Memory Management in Python, Garbage Collection in Python, Comparisons between C and Python, Comparisons between Java and Python.

Writing Our First Python Program: Installing Python for Windows, Installing numpy, Setting the Path to Python, Writing Our First Python Program, Executing a Python Program, Getting Help in Python, Getting Python Documentation Help, Reopening the Python Program in IDLE.

Data types in Python: Comments in Python, Doc strings, How Python Sees Variables, Data types in Python, Built-in data types, bool Data type, Sequences in Python, Sets, Literals in Python, Determining the Data type of a Variable, What about Characters, User-defined Data types, Constants in Python, Identifiers and Reserved words, Naming Conventions in Python.

Unit II

Operators in Python: Arithmetic Operators, Assignment Operators, Unary Minus Operator, Relational Operators, Logical Operators, Boolean Operators, Bitwise Operators, Membership Operators, Identity Operators, Operator Precedence and Associativity, Mathematical Functions.

Input and Output: Output statements, Input Statements, Command Line Arguments. Control Statements: Control Statements, The if Statement, A Word on Indentation, The if ... else Statement, The if ... elif ... else Statement, The while Loop, The for Loop, Infinite Loops, Nested Loops, The else Suite, The break Statement, The continue Statement, The pass Statement, The assert Statement, The return Statement.

Unit III

Arrays in Python: Array, Advantages of Arrays, Creating an Array, Importing the Array Module, Indexing and Slicing on Arrays, Processing the Arrays, Types of Arrays, Working with Arrays using numpy, Creating Arrays using array(), linspace, logspace, arange(), zeros() and ones() Functions, Mathematical Operations on Arrays, Comparing Arrays, Aliasing the Arrays, Viewing and Copying Arrays, Slicing and Indexing in numpy Arrays, Dimensions of Arrays, Attributes of an Array, The reshape() Method, The flatten() Method, Working with Multi-dimensional Arrays, Indexing in Multi-dimensional Arrays, Slicing the Multi-dimensional Arrays, Matrices in numpy, Getting Diagonal Elements of a Matrix, Finding Maximum and Minimum Elements, Finding Sum and Average of Elements, Products of Elements, Sorting the Matrix, Transpose of a Matrix, Matrix Addition and Multiplication, Random Numbers.

Strings and Characters: Creating Strings, Length of a String, Indexing in Strings, Slicing the Strings, Repeating the Strings, Concatenation of Strings, Checking Membership, Comparing Strings, Removing Spaces from a String, Finding Sub Strings, Counting Substrings in a String, Strings are Immutable, Replacing a String with another String, Splitting and Joining Strings, Changing Case of a String, Checking Starting and Ending of a String, String Testing Methods, Formatting the Strings,

Working with Characters, Sorting Strings, Searching in the Strings, Finding Number of Characters and Words, Inserting Sub String into a String.

Unit IV

Functions: Difference between a Function and a Method, Defining a Function, Calling a Function, Returning Results from a Function, Returning Multiple Values from a Function, Functions are First Class Objects, Pass by Object Reference, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Variable Length Arguments, Local and Global Variables, The Global Keyword, Passing a Group of Elements to a Function, Recursive Functions, Anonymous Functions or Lambdas, Function Decorators, Generators, Structured Programming, Creating our Own Modules in Python, The Special Variable name.

Lists and Tuples: List, Creating Lists using range() Function, Updating the Elements of a List, Concatenation of Two Lists, Repetition of Lists, Membership in Lists, Aliasing and Cloning Lists, Methods to Process Lists, Finding Biggest and Smallest Elements in a List, Sorting the List Elements, Number of Occurrences of an Element in the List, Finding Common Elements in Two Lists, Storing Different Types of Data in a List, Nested Lists, Nested Lists as Matrices, List Comprehensions, Tuples, Creating Tuples, Accessing the Tuple Elements, Basic Operations on Tuples, Functions to Process Tuples, Nested Tuples, Inserting Elements in a Tuple, Modifying Elements of a Tuple, Deleting Elements from a Tuple.

Text Book:

R. Nageswara Rao, Corer Python Programming, Dreamtech Press

References:

1. Mark Lutz, Learning Python
2. Tony Gaddis, Starting Out With Python
3. Kenneth A. Lambert, Fundamentals of Python
4. James Payne, Beginning Python using Python 2.6 and Python 3
5. Paul Gries, Practical Programming: An Introduction to Computer Science using Python 3

Note:

Student friendly video lecturers pertaining to this course are available at <http://spoken-tutorial.org/>

Teachers are advised to teach these courses in the computer lab itself, so that the interested students may derive some time to perform few programs their own.

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year
SEMESTER - V:

Elective 1

C) Python Lab

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
 - Faculty must take care about UG Standard Programs.
 - In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
 - External Vice-Voce is compulsory.
1. Write a Python Program which accepts the radius of a Circle from the user and compute the Area.
 2. Write a Python program to solve $(x + y) * (x + y)$, accept x and y from the user.
 3. Write a Python program to calculate the sum of three given numbers, if the values are equal then return thrice of their sum.
 4. Write a Python program to find whether a given number (accept from the user) is even or odd, Print out an appropriate message to the user.
 5. Write a Python program to get the Fibonacci series between 0 and 50.
 6. Write a Python program to print alphabet pattern 'E'.

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

7. Write the Python programs for the following.
 - a. To find the length of a given string
 - b. To Concatenate N Strings
 - c. To find the given String is palindrome or not.
8. Write a Python program to check if a String is numeric.
9. Write a Python program to find the sum of positive numbers from a given list (using continue statement).
10. Write a Python program to reverse the order of the items in the array.
11. Write a Python function to find the Maximum of three numbers.
12. Write a Python function that takes a number as a parameter and check the number is prime or not.
13. Write a Python program to get the number of occurrences of a specified element in an array.
14. Write a Python Recursive function to calculate the Factorial of a number (a non-negative number).
15. Write a Python program to add an item in a tuple.
16. Write a Python program to check whether an element exists within a tuple

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year (DSC-1F)
SEMESTER - VI

Elements of Scripting Languages

Unit I

HTML, Browsers and their types, URL's, web sites, Domain Names, static and dynamic sites and active web pages, Files Creation, Web Server, Web Client/Browser Hyper Text Markup Language, HTML Tags, Paired Tags, Commonly used HTML Commands Titles and Footers, Paragraph Breaks, Line Breaks, Heading Styles, Drawing Lines, Text Styles, Other Text Effects, Indenting Text, Lists, Types of Lists.

Unit II

Using the Border attribute, Using the Width and Height Attribute, Using the Align Attribute, Tables - Header, Data rows, The Caption Tag, Attributes - Width and Border, BGCOLOR, COLSPAN, ROWSPAN, External Document References, Internal Document References, Images as Hyperlinks, Introduction to Frames, tag, <FRAME> tag.

DHTML Introduction, use and its elements, Cascading Style Sheets – Introduction, Using Inline Styles, Sample Examples, Defining Your Own Styles, Properties in Values in Styles, A worked example, Formatting Blocks of Information, Layers, Embedded Style Sheets, Linking external sheets.

Unit III

JavaScript, Advantages, JavaScript Syntax, Data Types and Literal, Type Casting, Variables, Incorporating variables in a Script, Array, Operators and Expressions, Arithmetic Operators, Logical Operators, Comparison Operators, String Operators, Assignment Operators, Conditional Expression, Ternary and Special Operators

JavaScript Programming Constructs, If - then - else, Immediate If, For Loop, Built-in Functions, User Defined functions, Declaring functions, Place of Declaration, Passing Parameters, Variable Scope, Return Values, Recursive Functions, Placing text in a Browser, Dialog Boxes - Alert dialog box, Prompt dialog box, Confirm dialog box.

Unit IV

The Form Object, The Form Object's Methods, Text Element, Password Element, Button Element, Submit Button Element, Reset Button Element, Checkbox Element, Radio Element, Text Area Element, Select and Option Element, Multi Choice Select Lists Element, Form Validations using JavaScript.

Built-In Objects in JavaScript - String, Math, Date Object, Creating a User Defined Object.

Text Books:

1. Web Programming –Chris Bates – Third Edition.(Wiley)
2. Internet & World Wide Web- H. M. Deitel, P.J. Deitel, A. B. Goldberg-Third Edition

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year (DSC-1F)
SEMESTER - VI

Elements of Scripting Language Lab

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
 - Faculty must take care about UG Standard Programs.
 - In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
 - External Vice-Voce is compulsory.
1.
 - a. Write a HTML program using basic text formatting tags, <p>,
, <pre>.
 - b. Write a HTML page for Example Cafe using above text formatting tags.
 2.
 - a. Write a HTML program using presentational element tags , <i>, <strike>, <sup>, <sub>, <big>, <small>, <hr>
 - b. Write a HTML program using phrase element tags <blockquote>, <cite>, <abbr>, <acronym>, <kbd>, <address>
 3.
 - a. Write a HTML program using different list types.
 - b. Write a HTML page that displays ingredients and instructions to prepare a recipe.
 4.
 - a. Write a HTML program using grouping elements <div> and .
 - b. Write a HTML Menu page for Example cafe site.
 5.
 - a. Write a HTML program using images, audios, videos.
 - b. Write a HTML program to create your time table.
 6. Write a HTML program to create a form using text inputs, password inputs, multiple line text input, buttons, check boxes, radio buttons, select boxes, file select boxes.
 7. Write a HTML program to create frames and links between frames.
 8. Write a HTML program to create different types of style sheets.
 9. Write a HTML program to create CSS on links, lists, tables and generated content.
 10. Write a HTML program to create your college web site using multi column layouts.
 11. Write a HTML program to create your college web site using for mobile device.
 12. Write a HTML program to create login form and verify username and password.
 13.
 - a. Write a JavaScript program to calculate area of rectangle using function.
 - b. Write a JavaScript program to wish good morning, good afternoon, good evening depending on the current time.
 14.
 - a. Write a JavaScript program using switch case?
 - b. Write a JavaScript program to print multiplication table of given number using loop.
 15.
 - a. Write a JavaScript programs using any 5 events.
 - b. Write a JavaScript program using JavaScript built in objects.
 16. Write a JavaScript program to create registration Form with Validations.

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year (DSC-1F)
SEMESTER - VI

Elective 2

A) Operating Systems

Unit I

Introduction: Computer-System Architecture, Computing Environments.

Operating-System Structures: Operating-System Services, User Interface for Operating-System, System Calls, Types of System Calls, Operating System Structure.

Process Management: Process Concept, Process Scheduling, Operations on Processes, Inter process Communication, Examples–Producer-Consumer Problem.

Unit II

CPU Scheduling: Concepts, Scheduling Criteria, Scheduling Algorithms.

Process Synchronization: Critical-Section Problem, Peterson’s Solution, Synchronization, Semaphores, Monitors.

Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

Unit III

Main Memory: Introduction, Swapping, Contiguous Memory Allocation, Segmentation, Paging.

Virtual Memory: Introduction, Demand Paging, Page Replacement, Allocation of Frames, Thrashing.

Unit IV

Mass-Storage Structure: Overview, Disk Scheduling, RAID Structure.

File Systems: File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, Protection. File System Implementation, Directory Implementation, Allocation Methods, Free-Space Management.

Text Book:

Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Concepts (9e)

References:

1. Naresh Chauhan, Principles of Operating Systems
2. Thomas W. Doeppner, Operating Systems in Depth
3. Andrew S. Tanenbaum, Modern Operating Systems
4. William Stallings, Operating Systems – Internals and Design Principles
5. Dhananjay M. Dhandhere, Operating Systems – A Concept Based Approach

KAKATIYA UNIVERSITY
U.G. Computer Science (Under CBCS)
B.Sc. Final Year (DSE-1F)
SEMESTER – VI

A.) Operating Systems Lab

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
- Faculty must take care about UG Standard Programs.
- In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
- External Vice-Voce is compulsory.

1.
 - a) Use vi editor to create different files, writing data into files, modifying data in files.
 - b) Use different types of Unix commands on the files created in first program.
2. Write shell programs using 'case', 'then' and 'if' & 'else' statements.
3. Write shell programs using while, do-while and for loop statements.
4.
 - a) Write a shell script that accepts two integers as its arguments and compute the value of first number raised to the power of the second number.
 - b) Write a shell script that takes a command –line argument and reports on whether it is directory, a file, or something else.
5.
 - a) Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers..
 - b) Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
6.
 - a) Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
 - b) Develop an interactive script that ask for a word and a file name and then tells how many times that word occurred in the file.
7. Write a program to simulate the UNIX commands like ls, mv, cp.
8. Write a program to convert upper case to lower case letters of a given ASCII file.
9. Write a program to search the given pattern in a file.
10. Write a program to demonstrate FCFS process schedules on the given data.
11. Write a program to demonstrate SJF process schedules on the given data.
12. Write a program to demonstrate Priority Scheduling on the given burst time and arrival times.
13. Write a program to demonstrate Round Robin Scheduling on the given burst time and arrival times.
14. Write a program to implementing Producer and Consumer problem using Semaphores.
15. Write a program to simulate FIFO, LRU, LFU Page replacement algorithms.
16. Write a program to simulate Sequential, Indexed and Linked file allocation.

Programming in C++ Semester -II

Theory	4 Hours/Week	4 credits
Practical	3 Hours/Week	1 credit

Unit – I

Introduction to C++: Applications, Example Programs, Tokens, Data Types, Operators, Expressions, Control Structures, Arrays, Strings, Pointers, Searching and Sorting Arrays.
Functions: Introduction, Prototype, Passing Data by Value, Reference Variables, Using Reference Variables as Parameters, Inline Functions, Default Arguments, Overloading Functions, Passing Arrays to Functions.
Object Oriented Programming: Procedural and Object-Oriented Programming, Terminology, Benefits, OOP Languages, and OOP Applications.

Unit – II

Classes: Introduction, Defining an Instance of a Class, Why Have Private Members? Separating Class Specification from Implementation, Inline Member Functions, Constructors, Passing Arguments to Constructors, Destructors, Overloading Constructors, Private Member Functions, Arrays of Objects, Instance and Static Members, Friends of Classes, Member-wise Assignment, Copy Constructors, Operator Overloading, Object Conversion, Aggregation.

Unit – III


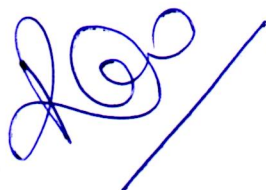

Inheritance: Introduction, Protected Members and Class Access, Base Class Access Specification, Constructors and Destructors in Base and Derived Classes, Redefining Base Class Functions, Class Hierarchies, Polymorphism and Virtual Member Functions, Abstract Base Classes and Pure Virtual Functions, Multiple Inheritance.
C++ Streams: Stream Classes, Unformatted I/O Operations, Formatted I/O Operations.

Unit – IV

Exceptions: Introduction, Throwing an Exception, Handling an Exception, Object-Oriented Exception Handling with Classes, Multiple Exceptions, Extracting Data from the Exception Class, Re-throwing an Exception, Handling the `bad_alloc` Exception.
Templates: Function Templates–Introduction, Function Templates with Multiple Type, Overloading with Function Templates, Class Templates – Introduction, Defining Objects of the Class Template, Class Templates and Inheritance, Introduction to the STL.

Text Tony Gaddis, Starting out with C++: from control structures through objects (7e)

References B. Lippman, C++ Primer
Bruce Eckel, Thinking in C++
K.R. Venugopal, Mastering C++
Herbert Schildt, C++: The Complete Reference
Bjarne Stroustrup, The C++ Programming Language
Sourav Sahay, Object Oriented Programming with C++



CHAIRMAN
Board of Studies
Department of Computer Science
KAKATIYA UNIVERSITY
VAREASIGUDA - 500002 (T.S.)

C++ Lab Semester -II

Practical

3 Hours/Week

1 credit

- 1 Write a program to.
 - a. Print the sum of digits of a given number.
 - b. Check whether the given number is Armstrong or not
 - c. Print the prime number from 2 to n where n is natural number given.
- 2 Write a program to find largest and smallest elements in a given list of numbers and sort the given list.
- 3 Write a program to read the student name, roll no, marks and display the same using class and object.
- 4 Write a program to implement the dynamic memory allocation and de-allocation using new and delete operators using class and object.
- 5 Write a program to find area of a rectangle, circle, and square using constructors.
- 6 Write a program to implement copy constructor.
- 7 Write a program using friend functions and friend class.
- 8 Write a program to implement constructors
 - § Default Constructor, Parameterized Constructor, Copy Constructor
 - § Define the constructor inside/outside of the class
 - § Implement all three constructors within a single class as well as use multiple classes(individual classes)Write a program to implement the following concepts using class and object
 - § Function overloading
 - § Operator overloading (unary/binary(+ and -))Write a program to demonstrate single inheritance, multilevel inheritance and multiple inheritances.
Write a program to implement the overloaded constructors in inheritance.
Write a program to implement the polymorphism and the following concepts using class and object.
 - § Virtual functions
 - § Pure virtual functionsWrite a program to implement the virtual concepts for following concepts
 - § Constructor (not applied)
 - § Destructor (applied)Write a program to demonstrate static polymorphism using method overloading.
Write a program to demonstrate dynamic polymorphism using method overriding and dynamic method dispatch.
Write a program to implement the template (generic) concepts
 - § Without template class and object
 - § With template class and object

Write the Pseudo Code and draw Flow Chart for the above programs.

Recommended to use Open Source Software: GCC on Linux; DevC++ (or) CodeBlocks on Windows.

CHAIRMAN
Board of Studies
Department of Computer Science
KAKATIYA UNIVERSITY
VARANASI, UTTAR PRADESH (U.P.)

Programming in C Semester -I

Theory	4 Hours/Week	4 credit
Practical	3 Hours/Week	1 credit

Unit – I

Computer Fundamentals: Introduction of Computers, Classification of Computers, Anatomy of a Computer, Memory Hierarchy, Introduction to OS, Operational Overview of a CPU.
Program Fundamentals: Generation and Classification of Programming Languages, Compiling, Interpreting, Loading, Linking of a Program, Developing Program, Software Development.
Algorithms: Definitions, Different Ways of Stating Algorithms (Step-form, Pseudo-code, Flowchart), Strategy for Designing Algorithms, Structured Programming Concept.
Basics of C: Overview of C, Developing Programs in C, Parts of Simple C Program, Structure of a C Program, Comments, Program Statements, C Tokens, Keywords, Identifiers, Data Types, Variables, Constants, Operators and Expressions, Expression Evaluation—precedence and associativity, Type Conversions.

Unit – II

Input-Output: Non-formatted and Formatted Input and Output Functions, Escape Sequences,
Control Statements: Selection Statements – if, if-else, nested if, nested if-else, comma operator, conditional operator, switch; Iterative Statements—while, for, do-while; Special Control Statement—goto, break, continue, return, exit.
Arrays and Strings: One-dimensional Arrays, Character Arrays, Functions from ctype.h, string.h, Multidimensional Arrays.

Unit – III

Functions: Concept of Function, Using Functions, Call-by-Value Vs Call-by-reference, Passing Arrays to Functions, Scope of Variables, Storage Classes, Inline Functions, and Recursion.
Pointers: Introduction, Address of Operator (&), Pointer, Uses of Pointers, Arrays and Pointers, Pointers and Strings, Pointers to Pointers, Array of Pointers, Pointer to Array, Dynamic Memory Allocation.


Unit – IV

User-defined Data Types: Declaring a Structure (Union) and its members, Initialization Structure (Union), Accessing members of a Structure (Union), Array of Structures (Union), Structures versus Unions, Enumeration Types.
Files: Introduction, Using Files in C, Working with Text Files, Working with Binary Files, Files of Records, Random Access to Files of Records, Other File Management Functions.

Text Pradip Dey, Manas Ghosh, Computer Fundamentals and Programming in C (2e)

References BOOKS

Ivor Horton, Beginning C
Ashok Kamthane, Programming in C
Herbert Schildt, The Complete Reference C
Paul Deitel, Harvey Deitel, C How To Program
Byron S. Gottfried, Theory and Problems of Programming with C
Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language
B. A. Forouzan, R. F. Gilberg, A Structured Programming Approach Using C



CHAIRMAN
Board of Studies
Department of Computer Science
KAKATIYA UNIVERSITY
WARANGAL-506002 (T.S.)

With Effect from the Academic Year 2019-2020

C Lab Semester -I

Practical

3 Hours/Week

1 credit


- 1 Write a program to find the largest two (three) numbers using if and conditional operator.
- 2 Write a program to print the reverse of a given number.
- 3 Write a program to print the prime number from 2 to n where n is given by user.
- 4 Write a program to find the roots of a quadratic equation using switch statement.
- 5 Write a program to print a triangle of stars as follows (take number of lines from user):

```
      *
     ***
    *****
   *****
  *****
```
- 6 Write a program to find largest and smallest elements in a given list of numbers.
- 7 Write a program to find the product of two matrices..
- 8 Write a program to find the GCD of two numbers using iteration and recursion.
- 9 Write a program to illustrate use of storage classes.
- 10 Write a program to demonstrate the call by value and the call by reference concepts.
- 11 Write a program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
- 12 Write a program to illustrate use of data type enum.
- 13 Write a program to demonstrate use of string functions string.h header file.
- 14 Write a program that opens a file and counts the number of characters in a file.
- 15 Write a program to create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.
- 16 Write a program that opens an existing text file and copies it to a new text file with all lowercase letters changed to capital letters and all other characters unchanged.

Note

Write the Pseudo Code and draw Flow Chart for the above programs.
Recommended to use Open Source Software: GCC on Linux; DevC++ (or) CodeBlocks on Windows 10.




CHAIRMAN
Board of Studies
Department of Computer Science
KAKATIYA UNIVERSITY
WARRANGAL-500009 (T.S.)



KAKATIYA UNIVERSITY

Under Graduate Courses (Under CBCS 2020–2021 onwards)

B.Sc. Computer Science II Year SEMESTER – III

DATA STRUCTURES USING C++

Theory: 4 Hours/Week; **Credits:** 4 **Marks:** 100 (Internal: 20; External: 80)
Practical: 3 Hours/Week **Credits:** 1 **Marks:** 25

Unit - I

Basic data Structure: Introduction to Data Structures, Types of Data Structures, and Introduction to Algorithms, Pseudo code, and Relationship among data, data structures, and algorithms, Implementation of data structures, Analysis of Algorithms.

Stacks: Concept of Stacks and Queues, Stacks, Stack Abstract Data Type, Representation of Stacks Using Sequential Organization (Arrays), Multiple Stacks, Applications of Stack, Expression Evaluation and Conversion, Polish notation and expression conversion, Processing of Function Calls, Reversing a String with a Stack, Recursion.

Unit - II

Recursion: Introduction, Recurrence, Use of Stack in Recursion, Variants of Recursion, Recursive Functions, Iteration versus Recursion.

Queues: Concept of Queues, Queue as Abstract Data Type, Realization of Queues Using Arrays, Circular Queue, Multi-queues, Dequeue, Priority Queue, Applications of Queues,

Linked Lists: Introduction, Linked List, Linked List Abstract Data Type, Linked List Variants, Doubly Linked List, Circular Linked List, Representation of Sparse Matrix Using Linked List, Linked Stack, Linked Queue.

Unit - III

Trees: Introduction, Types of Trees, Binary Tree, Binary Tree Abstract Data Type, Realization of a Binary Tree, Insertion of a Node in Binary Tree, Binary Tree Traversal, Other Tree Operations, Binary Search Tree, Threaded Binary Tree, Applications of Binary Trees.

Searching and Sorting: Search Techniques-Linear Search, Binary Search, Sorting Techniques- Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort, Comparison of All Sorting Methods, Search

Trees: Symbol Table, Optimal Binary Search Tree, AVL Tree (Height-balanced Tree).

Unit - IV

Graphs: Introduction, Representation of Graphs, Graph Traversal – Depth First Search, Breadth First Search, Spanning Tree, Prim's Algorithm, Kruskal's Algorithm.

Hashing: Introduction, Key Terms and Issues, Hash Functions, Collision Resolution Strategies, Hash Table Overflow, Extendible Hashing

Heaps: Basic Concepts, Implementation of Heap, Heap as Abstract Data Type, Heap Sort, Heap Applications.

Text books:

1. Varsha H. Patil "Data structures using C++" Oxford University press, 2012
2. M.T. Goodrich, R. Tamassia and D. Mount, Data Structures and Algorithms in C++, John Wiley and Sons, Inc., 2011.

References:

1. Adam Drozdek "Data structures and algorithm in C++" Second edition, 2001
2. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, Introduction to Algorithms, 2nd Ed., Prentice-Hall of India, 2006.
3. Robert L. Kruse and A.J. Ryba, Data Structures and Program Design in C++, PrenticeHall, Inc., NJ, 1998.
4. B. Stroustrup, The C++ Programming Language, Addison Wesley, 2004
5. D.E. Knuth, Fundamental Algorithms (Vol. I), Addison Wesley, 1997

KAKATIYA UNIVERSITY
Under Graduate Courses (Under CBCS 2020–2021 onwards)
B.Sc. Computer Science II Year
SEMESTER – III

DATA STRUCTURES USING C++ LAB
Practical 3 Hours/Week 1 Credit Marks: 25

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
 - In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
 - External Vice-Voce is compulsory.
1. Write C++ programs to implement the following using an array
 - a) Stack ADT
 - b) Queue ADT
 2. Write a C++ program to implement Circular queue using array.
 3. Write C++ programs to implement the following using a single linked list.
 - a) Stack ADT
 - b) Queue ADT
 4. Write a C++ program to implement Circular queue using Single linked list.
 5. Write a C++ program to implement the double ended queue ADT using double linked list.
 6. Write a C++ program to solve tower of Hanoi problem recursively
 7. Write C++ program to perform the following operations:
 - a) Insert an element into a binary search tree.
 - b) Delete an element from binary search tree.
 - c) Search for a key in a binary search tree.
 8. Write C++ programs for the implementation tree traversal technique BFS.
 9. Write a C++ program that uses recursive functions to traverse a binary search tree.
 - a) Pre-order
 - b) In-order
 - c) Post-order
 10. Write a C++ program to find height of a tree.
 - 11 Write a C++ program to find MIN and MAX element of a BST.
 - 12 Write a C++ program to find Inorder Successor of a given node.
 13. Write C++ programs to perform the following operations on B-Trees and AVL Trees.
 - a) Insertion
 - b) Deletion
 - 14 Write C++ programs for sorting a given list of elements in ascending order using the following sorting methods.
 - a) Quick sort
 - b) Merge sort
 15. Write a C++ program to find optimal ordering of matrix multiplication.
 16. Write a C++ program that uses dynamic programming algorithm to solve the optimal binary search tree problem
 17. Write a C++ program to implement Hash Table
 18. Write C++ programs to perform the following on Heap
 - a) Build Heap
 - b) Insertion
 - c) Deletion
 19. Write C++ programs to perform following operations on Skip List
 - a) Insertion
 - b) Deletion
 20. Write a C++ Program to Create a Graph using Adjacency Matrix Representation.
 21. Write a C++ program to implement graph traversal techniques
 - a) BFS
 - b) DFS
 22. Write a C++ program to Heap sort using tree structure.

KAKATIYA UNIVERSITY

Under Graduate Courses (Under CBCS 2020 – 2021 onwards)

B.Sc. Computer Science II Year

SEMESTER – IV

DATA BASE MANAGEMENT SYSTEMS

Theory: 4 Hours/Week; Credits: 4 Marks: 100 (Internal: 20; External: 80)

Practical: 3 Hours/Week Credits: 1 Marks: 25

Unit - I

Introduction: Database-System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Database Users and Administrators.

Introduction to the Relational Model: Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relational Query Languages, Relational Operations.

Unit - II

Database Design and the E-R Model: Overview of the Design Process, The Entity- Relationship Model, Constraints, Removing Redundant Attributes in Entity Sets, Entity-Relationship Diagrams, Reduction to Relational Schemas, Entity-Relationship Design Issues, Extended E-R Features, Alternative Notations for Modeling Data, Other Aspects of Database Design.

Relational Database Design: Features of Good Relational Designs, Atomic Domains and First Normal Form, Decomposition Using Functional Dependencies, Functional- Dependency Theory, Decomposition Using Multivalued Dependencies, Normal Forms-2 NF, 3 NF, BCNF, The Database Design Methodology for Relational Databases.

Unit - III

Introduction to SQL: Overview of the SQL Query Language, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Subqueries, Modification of the Database.

Intermediate SQL: Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization.

Advanced SQL: Accessing SQL from a Programming Language, Functions and Procedures, Triggers, Recursive Queries.

Unit - IV

Transaction Management: Transaction Support–Properties of Transactions, Database Architecture, Concurrency Control–The Need for Concurrency Control, Serializability and Recoverability, Locking Methods, Deadlock, Time Stamping Methods, Multi-version Timestamp Ordering, Optimistic Techniques, Granularity of Data Items, Database Recovery–The Need for Recovery, Transactions and Recovery, Recovery Facilities, Recovery Techniques, Nested Transaction Model. Security: Database Security–Threats, Computer-Based Controls–Authorization, Access Controls, Views, Backup and Recovery, Integrity, Encryption, RAID.

Text book:

1. Silberschatz, H. Korth and S. Sudarshan, Database System Concepts, 6th Ed., Tata McGraw Hill, 2011
2. Thomas M. Connolly, Carolyn E. Begg, Database Systems–A Practical Approach to Design, Implementation, and Management (6e)

KAKATIYA UNIVERSITY

Under Graduate Courses (Under CBCS 2020 – 2021 onwards)

B.Sc. Computer Science II Year

SEMESTER – IV

DATA BASE MANAGEMENT SYSTEMS - LAB

Practical

3 Hours/Week

1 Credit

Marks: 25

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
- In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
- External Vice-Voce is compulsory.

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

LibraryBooks (Accession number, Title, Author, Department, PurchaseDate, Price),

IssuedBooks (Accession number, 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 “CS” department and are written by author “Navathe”.
- f) List all computer (Department=“CS”) that have been issued.
- g) List all books which have a price less than 500 or purchased between “01/01/1999” and “01/01/2004”.

2. Create a database having three tables to store the details of students of Computer Department in your college.

Personal information about Student (College roll number, Name of student, Date of birth, Address, Marks(rounded off to whole number) in percentage at 10 + 2, Phone number)

Paper Details (Paper code, Name of the Paper)

Student’s Academic and Attendance details (College roll number, Paper Code, Attendance, Marks in home examination).

- a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.
- b) Design a query that will return the records (from the second table) along with the name of student from the first table, related to students who have more than 75% attendance and more than 60% marks in paper2.
- c) List all students who live in “Warangal” and have marks greater than 60 in paper1.
- d) Find the total attendance and total marks obtained by each student.
- e) List the name of student who has got the highest marks in paper2.

3. Create the following tables and answer the queries given below:

Customer (CustID, email, Name, Phone, ReferrerID)

Bicycle (BicycleID, DatePurchased, Color, CustID, ModelNo)

BicycleModel(ModelNo, Manufacturer, Style) Service

(StartDate, BicycleID, EndDate)

- a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.
- b) List all the customers who have the bicycles manufactured by manufacturer "Honda".
- c) List the bicycles purchased by the customers who have been referred by Customer "C1".
- d) List the manufacturer of red colored bicycles.
- e) List the models of the bicycles given for service.

4. Create the following tables, 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 \$10,000.
- 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.

5. 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, Qunatity)

- a) Identify primary and foreign keys.
- b) Get supplier numbers for suppliers in Paris with status>20.
- c) Get suppliers details for suppliers who supply part P2. Display the supplier list in increasing order of supplier numbers.
- d) Get suppliers names for suppliers 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 part nos. for parts that either weigh more than 16 pounds or are supplied by suppliers S2, or both.
- h) Get the names of cities that store more than five red parts.
- i) Get full details of parts supplied by a supplier in Hyderabad.

- j) Get part numbers for part supplied by a supplier in Warangal to a project in Chennai.
 - k) Get the total number of project supplied by a supplier (say, S1).
 - l) Get the total quantity of a part (say, P1) supplied by a supplier (say, S1).
-
- 6. Write a PL/SQL Program to demonstrate Procedure.
 - 7. Write a PL/SQL Program to demonstrate Function.
 - 8. Write a PL/SQL program to Handle Exceptions.
 - 9. Write a PL/SQL Program to perform a set of DML Operations.
 - 10. Create a View using PL/SQL program.
 - 11. Write a PL/SQL Program on Statement Level Trigger.
 - 12. Write a PL/SQL Program on Row Level Trigger.

KAKATIYA UNIVERSITY
FACULTY OF SCIENCE
B.Sc. (Computer Science)
SEMESTER – V
Programming in Java

Theory	4 Hours/Week	4 Credit	Internal marks = 20
Practical	3 Hours/Week	1 Credit	External Marks = 80

Unit - I

Introduction: Java Essentials, JVM, Java Features, Creation and Execution of Programs, Data Types, Structure of Java Program, Type Casting, Conditional Statements, Loops, Classes, Objects, Class Declaration, Creating Objects.

Unit - II

Method Declaration and Invocation, Method Overloading, Constructors – Parameterized Constructors, Constructor Overloading, Cleaning-up unused Objects. Class Variables & Method-static Keyword, this Keyword, One-Dimensional Arrays, Two-Dimensional Arrays, Command-Line Arguments, Inner Class.

Inheritance: Introduction, Types of Inheritance, extends Keyword, Examples, Method Overriding, super, final Keyword, Abstract classes, Interfaces, Abstract Classes Verses Interfaces.

Packages: Creating and Using Packages, Access Protection, Wrapper Classes, String Class, StringBuffer Class.

Unit - III

Exception: Introduction, Types, Exception Handling Techniques, User-Defined Exception.

Multithreading: Introduction, Main Thread and Creation of New Threads –By Inheriting the Thread Class or Implementing the Runnable Interface, Thread Lifecycle, Thread Priority and Synchronization.

Input/Output: Introduction, java.io Package, File Streams, FileInputStream Class, FileOutputStream Class, Scanner Class, BufferedInputStream Class, BufferedOutputStream Class, RandomAccessFile Class.

Unit - IV

Applets: Introduction, Example, Life Cycle, Applet Class, Common Methods Used in Displaying the Output (Graphics Class).

Event Handling: Introduction, Types of Events, Example.

AWT: Introduction, Components, Containers, Button, Label, Checkbox, Radio Buttons, Container Class, Layouts.

Swings: Introduction, Differences between Swing and AWT, JFrame, JApplet, JPanel, Components in Swings, Layout Managers, JTable.

Text Book:


1. Sachin Malhotra, Saurabh Choudhary, Programming in Java (2e)

References:

1. Bruce Eckel, Thinking in Java (4e)
2. Herbert Schildt, Java: The Complete Reference (9e)
3. Y. Daniel Liang, Introduction to Java Programming (10e)
4. Paul Deitel, Harvey Deitel, Java: How To Program (10e)
5. Cay S. Horstmann, Core Java Volume I –Fundamentals (10e)

Department of Computer Science, KU

With Effect from the Academic Year 2019-2020


D. S. RAMIA
Chairperson BOS
Department of Computer Science
KAKATIYA UNIVERSITY
Warangal- 506 009 (T.S.)

KAKATIYA UNIVERSITY
FACULTY OF SCIENCE
B.Sc. (Computer Science)
SEMESTER – V
Programming in Java Lab

Practical 3 Hours/Week 1 Credit Marks: 25

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
 - In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
 - External Vice-Voce is compulsory.
1. Write a program to find the largest of n natural numbers.
 2. Write a program to find whether a given number is prime or not.
 3. Write a menu driven program for following:
 - a. Display a Fibonacci series
 - b. Compute Factorial of a number
 4. Write a program to check whether a given number is odd or even.
 5. Write a program to check whether a given string is palindrome or not.
 6. Write a program to print the sum and product of digits of an Integer and reverse the Integer.
 7. Write a program to create an array of 10 integers. Accept values from the user in that Array. Input another number from the user and find out how many numbers are equal to the number passed, how many are greater and how many are less than the number passed.
 8. Write a program that will prompt the user for a list of 5 prices. Compute the average of the prices and find out all the prices that are higher than the calculated average.
 9. Write a program in java to input N numbers in an array and print out the Armstrong numbers from the set.
 10. Write java program for the following matrix operations:
 - a. Addition of two matrices
 - b. Transpose of a matrix
 11. Write a java program that computes the area of a circle, rectangle and a Cylinder using function overloading.
 12. Write a Java program for the implementation of multiple inheritance using interfaces to calculate the area of a rectangle and triangle.
 13. Write a java program to create a frame window in an Applet. Display your name, address and qualification in the frame window.
 14. Write a java program to draw a line between two coordinates in a window.
 15. Write a java program to display the following graphics in an applet window.
 - a. Rectangles b. Circles
 - c. Ellipses d. Arcs e. Polygons
 16. Write a program that reads two integer numbers for the variables a and b. If any other character except number (0-9) is entered then the error is caught by NumberFormatException object. After that ex.getMessage () prints the information about the error occurring causes.
 17. Write a program for the following string operations:
 - a. Compare two strings b. concatenate two strings c. Compute length of a string
 18. Create a class called Fraction that can be used to represent the ratio of two integers. Include appropriate constructors and methods. If the denominator becomes zero, throw and handle an exception.

Department of Computer Science, KU

With Effect from the Academic Year 2019-2020


Department of Computer Science
KAKATIYA UNIVERSITY
Warangal- 506 009 (T.S.)

KAKATIYA UNIVERSITY
FACULTY OF SCIENCE
B.Sc. (Computer Science)
SEMESTER – VI
Web Technologies

Theory	4 Hours/Week	4 Credit	Internal marks = 20
Practical	3 Hours/Week	1 Credit	External Marks = 80

Unit – I

Introduction To XHTML– Introduction, first HTML, Headings, Linking, Images, special characters and horizontal rules, Lists, Tables, Frames, Forms, internal linking, meta Elements. CASCADING STYLE SHEETS – Introduction, Inline Styles, Embedded Style Sheets, Conflicting Styles, Linking external sheets, position Elements, box model and text flow, media types, building a CSS drop-down menu, user style sheets, CSS3.

Unit – II

Introduction To Java Scripting- introduction, simple program, prompt dialog and alert boxes, memory concepts, operators, decision making, control structures, if... else statement, while, counter-controlled repetitions, switch statement, do... while statement, *break* and *continue* statements. Functions – program modules in JavaScript, programmer–defined functions, functions definition, scope rules, global functions, Recursion.

Unit – III

Arrays- introduction, declaring and allocating arrays, references and reference parameters, passing arrays to functions. Multidimensional arrays, **EVENTS** – registering event handling, event onload, onmouseover, onmouseout, onfocus, onblur, onsubmit, onreset, event bubbling, more events. **JAVA SCRIPT OBJECTS** – introduction to object technology, Math Object, String Object, Date Object, Boolean and Number Object, document and window Objects, using cookies.

Unit – IV

XML - Introduction, XML Basics, Structuring Data, XML Namespaces, Document Type Definitions (DTDs), W3C XML Schema Documents, XML Vocabularies, Extensible Style sheet Language and XSL Transformations, Document Object Model (DOM).

Ajax-Enabled Rich Internet Applications: introduction, history of Ajax, traditional web applications Vs Ajax Applications, RIAs with Ajax, Ajax example using XMLHttpRequest object, XML and DOM, creating full scale Ajax-enabled application, Dojo Toolkit.

Text Book:

1. Internet & World Wide Web: HOW TO PROGRAM- H. M. Deitel, P.J. Deitel, -Fourth Edition- Pearson edition.

Department of Computer Science, KU

With Effect from the Academic Year 2019-2020


Dr. B. RAMA
Department of Computer Science
KAKATIYA UNIVERSITY
Warangal- 506 009 (T.S.)

KAKATIYA UNIVERSITY
FACULTY OF SCIENCE
B.Sc. (Computer Science)
SEMESTER – VI
Web Technologies Lab

Practical 3 Hours/Week 1 Credit Marks: 25

1. Write a HTML program using basic text formatting tags, <p>,
, <pre>.
2. Write a HTML program by using text formatting tags.
3. Write a HTML program using presentational element tags , <i>, <strike>, <sup>, <sub>, <big>, <small>, <hr>
4. Write a HTML program using phrase element tags <blockquote>, <cite>, <abbr>, <acronym>, <kbd>, <address>
5. Write a HTML program using different list types.
6. Create a HTML page that displays ingredients and instructions to prepare a recipe.
7. Write a HTML program using grouping elements <div> and .
8. Write a HTML Menu page for Example cafe site.
9. Write a HTML program using images, audios, videos.
10. Write a HTML program to create your time table.
11. Write a HTML program to create a form using text inputs, password inputs, multiple line text input, buttons, check boxes, radio buttons, select boxes, file select boxes.
12. Write a HTML program to create frames and links between frames.
13. Write a HTML program to create different types of style sheets.
14. Write a HTML program to create CSS on links, lists, tables and generated content.
15. Write a HTML program to create your college web site using multi column layouts.
16. Write a HTML program to create your college web site using for mobile device.
17. Write a HTML program to create login form and verify username and password.
18. Write a JavaScript program to calculate area of rectangle using function.
19. Write a JavaScript program to wish good morning, good afternoon, good evening depending on the current time.
20. Write a JavaScript program using switch case?
21. Write a JavaScript program to print multiplication table of given number using loop.
22. Write a JavaScript programs using any 5 events.
23. Write a JavaScript program using JavaScript built in objects.
24. Write a JavaScript program to create registration Form with Validations.
25. Write a XML Program to represent Student Data using DTD.
26. Write a XML Program to represent Data using XML Schema Definition.