

TELANGANA STATE COUNCIL OF HIGHER EDUCATION

B.Sc. (Computer Science) Course Structure

(Common Core Syllabus for All Universities of Telanagana State for the Students Admitted from the Academic Year: 2019-20 Batch Onwards)

CBCS Pattern in Semester System – 2019

Paper	Semester	Course Title	Hours / week		Credits
			Theory	Practical	
DSC – I	I	Programming in C	4	3	4 + 1 = 5
DSC – II	II	Programming in C++	4	3	4 + 1 = 5
DSC – III	III	Data Structures Using C++	4	3	4 + 1 = 5
DSC – IV	IV	Data Base Management Systems (DBMS)	4	3	4 + 1 = 5
DSC – V	V	Programming in Java	4	3	4 + 1 = 5
DSC – VI	VI	Web Technologies	4	3	4 + 1 = 5

Paper	Semester	Course Title	Hours / week		Credits
			Theory	Practical	
SEC – I	III	Python – I	2		2
SEC – II	III	Operating Systems - I	2		2
SEC – III	IV	Python – II	2		2
SEC – IV	IV	Operating Systems - II	2		2

Paper	Semester	Course Title	Hours / week		Credits
			Theory	Practical	
AECC	I	Fundamentals of Computer	2		2
AECC	II	Office Automation	2		2

Paper	Semester	Course Title	Hours / week		Credits
			Theory	Practical	
GENERAL ELECTIVE (GE)	IV	Information Technologies	4		4

PROJECT / OPTINAL

Paper	Semester	Course Title	Hours / week		Credits
			Theory	Practical	
PROJECT / OPTINAL	VI	PHP with MYSQL	3	3	3 + 1 = 4

Theory: 4 Hrs/Wk (4 Credits)

Practical: 3 Hrs/Wk (1 Credit)

Total Credits: 4+1 = 5 Credits

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

Reference Books:

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

C – Lab

Practical: 3 Hrs/Wk (1 Credits)

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:

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

Programming in C++ (Semester – II)

Theory: 4 Hrs/Wk (4 Credits)

Practical: 3 Hrs/Wk (1 Credit)

Total Credits: 4+1 = 5 Credits

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

References:

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

C++ – Lab

Practical: 3 Hrs/Wk (1 Credits)

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

Note:

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

Data Structures using C++ (Semester – III)

Theory: 4 Hrs/Wk (4 Credits)

Practical: 3 Hrs/Wk (1 Credit)

Total Credits: 4+1 = 5 Credits

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 Book:

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

Data Structures using C++ – Lab

Practical: 3 Hrs/Wk (1 Credits)

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.

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.

Data Base Management Systems (Semester – IV)

Theory: 4 Hrs/Wk (4 Credits)

Practical: 3 Hrs/Wk (1 Credit)

Total Credits: 4+1 = 5 Credits

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)

Data Base Management Systems – Lab

Practical: 3 Hrs/Wk (1 Credits)

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.

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.

Programming in Java (Semester – V)

Theory: 4 Hrs/Wk (4 Credits)

Practical: 3 Hrs/Wk (1 Credit)

Total Credits: 4+1 = 5 Credits

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, SaurabhChoudhary, 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)

Programming in Java – Lab

Practical: 3 Hrs/Wk (1 Credits)

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.

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.

Theory: 4 Hrs/Wk (4 Credits)

Practical: 3 Hrs/Wk (1 Credit)

Total Credits: 4+1 = 5 Credits

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.

Web Technologies – Lab

Practical: 3 Hrs/Wk (1 Credits)

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.

Fundamental of Computers (Semester – I)

Theory: 2 Hrs/Wk

AECC

Total Credits: 2 Credits

UNIT-I

Introduction to Computers: what is a computer, characteristics of Computers, Generations of Computers, Classifications of Computers, Basic Computer organization, Applications of Computers. Input and Output Devices: Input devices, Output devices, Softcopy devices, Hard copy devices. Computer Memory and Processors: Introduction, Memory Hierarchy, Processor, Registers, Cache memory, primary memory, secondary storage devices, magnetic tapes, floppy disks, hard disks, optical drives, USB flash drivers, Memory cards, Mass storage devices, Basic processors architecture.

UNIT-II

Number System and Computer Codes: Binary number system, working with binary numbers, octal number system, hexadecimal number system, working with fractions, signed number representation in binary form, BCD code, other codes. Boolean algebra and logic gates: Boolean algebra, Venn diagrams, representation of Boolean functions, logic gates, logic diagrams and Boolean expressions using karnaugh map. Computer Software: Introduction to computer software, classification of computer software, system software, application software, firmware, middleware, acquiring computer software, design and implementation of correct, efficient and maintainable programs.

Text Book:

Reema Thareja, Fundamentals of Computers.

References:

1. V.Rajaraman, 6th Edition Fundamentals of Computers, Neeharika Adabala.
2. Anita Goel, Computer Fundamentals.

Office Automation (Semester – II)

Theory: 2 Hrs/Wk

AECC

Total Credits: 2 Credits

UNIT-I

MS-Office: Introduction to MS-Office, MS-Word: Word Basics, working with formatted text, Shortcut keys, Formatting documents: Selecting text, Copying & moving data, Formatting characters, changing cases, Paragraph formatting, Indents, Drop Caps, Using format painter, Page formatting, Header & footer, Bullets & numbering, Tabs, Forming tables. Finding & replacing text, go to (F5) command, proofing text (Spell check, Auto correct), Reversing actions, Macros, Inserting pictures, Hyperlinks, Mail merging, Printing documents.

UNIT-II

Spreadsheets, workbooks, creating, saving & editing a workbook, Renaming sheet, cell entries (numbers, labels, and formulas), spell check, find and replace, Adding and deleting rows and columns Filling series, fill with drag, data sort, Formatting worksheet, Functions and its parts, Some useful Functions in excel (SUM, AVERAGE, COUNT, MAX, MIN, IF), Cell referencing (Relative, Absolute, Mixed), What-if analysis Introduction to charts: types of charts, creation of charts, printing a chart, printing worksheet.

Presentation Software, Uses, Presentation tips, components of slide, templates and wizards, using template, choosing an auto layout, using outlines, adding subheadings, editing text, formatting text, using master slide, adding slides, changing color scheme, changing background and shading, adding header and footer, adding clip arts and auto shapes. Various presentation, Working in slide sorter view (deleting, duplicating, rearranging slides), adding transition and animations to slide show, inserting music or sound on a slide, viewing slide show, Printing slides.

Text Book:

1. Computer Fundamentals – P.K. Sinha.
2. Introduction to Computers – N. Subramanian.
3. Introduction to Computers – Peter Norton Mcgraw Hill.
4. Fundamentals of Computers, by RemaTharaja, Oxford University Press India, 2014.
5. MS–Office _ BPB Publications.

Python - I (Semester – III)

Theory: 2 Hrs/Wk

SEC – I

Total Credits: 2 Credits

UNIT – I

Introduction to Python Programming: How a Program Works, Using Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations (Operators. Type conversions, Expressions), More about Data Output. Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables. Repetition Structures: Introduction, while loop, for loop, Calculating a Running Total, Input Validation Loops, Nested Loops.

UNIT – II

Functions: Introduction, Defining and Calling a Void Function, Designing a Program to Use Functions, Local Variables, Passing Arguments to Functions, Global Variables and Global Constants, Value-Returning Functions- Generating Random Numbers, Writing Our Own Value-Returning Functions, The math Module, Storing Functions in Modules. File and Exceptions: Introduction to File Input and Output, Using Loops to Process Files, Processing Records, Exceptions.

Text Book:

Tony Gaddis, Starting Out With Python (3e)

References:

1. Kenneth A. Lambert, Fundamentals of Python
2. Clinton W. Brownley, Foundations for Analytics with Python
3. James Payne, Beginning Python using Python 2.6 and Python 3
4. Charles Dierach, Introduction to Computer Science using Python
5. Paul Gries, Practical Programming: An Introduction to Computer Science using Python 3

Operating Systems - I (Semester – III)

Theory: 2 Hrs/Wk

SEC – II

Total Credits: 2 Credits

UNIT - I

Operating System: Introduction, The Operating System As A Resource Manager, History Of Operating Systems, The Operating System Zoo(Classifications), Operating System Concepts, System Calls, Operating System Structure(Architecture).

PROCESS: Creation, Hierarchies, States, THREADS- Usage, The Classical Thread Model, POSIX Threads, Pop-Up Threads.

UNIT - II

Scheduling: Introduction, Scheduling in Batch Systems, Scheduling in Interactive Systems, Scheduling in Real-Time Systems, Policy versus Mechanism, Thread Scheduling.

Memory Management: No Memory Abstraction, Memory Abstraction-Address Spaces, Virtual Memory, Page Replacement Algorithms, Design Issues for Paging Systems, Segmentation.

Text book:

1. A.S. Tanenbaum, and Herbert Bos, “Modern Operating Systems”, 4th Ed., Prentice-Hall of India, 2015.

References:

1. William Stallings, Operating Systems: Internals and Design Principles, 5th Ed., Prentice-Hall of India, 2006.
2. Gary Nutt, Operating Systems: A Modern Approach, 3rd Ed., Addison Wesley, 2004.
3. D.M. Dhamdhere, Operating Systems: A Concept Based Approach, 2nd Ed., Tata McGraw-Hill, 2007.

Python - II (Semester – IV)

Theory: 2 Hrs/Wk

SEC – III

Total Credits: 2 Credits

UNIT – I

Lists and Tuples: Sequences, Introduction to Lists, List slicing, Finding Items in Lists with the in Operator, List Methods and Useful Built-in Functions, Copying Lists, Processing Lists, Two-Dimensional Lists, Tuples. **Strings:** Basic String Operations, String Slicing, Testing, Searching, and Manipulating Strings. **Dictionaries and Sets:** Dictionaries, Sets, Serializing Objects.

Recursion: Introduction, Problem Solving with Recursion, Examples of Recursive Algorithms.

UNIT – II

Object-Oriented Programming: Procedural and Object-Oriented Programming, Classes, Working with Instances, Techniques for Designing Classes, Inheritance, Polymorphism.

GUI Programming: Graphical User Interfaces, Using the tkinter Module, Display text with Label Widgets, Organizing Widgets with Frames, Button Widgets and Info Dialog Boxes, Getting Input with Entry Widget, Using Labels as Output Fields, Radio Buttons, Check Buttons.

Text Book:

Tony Gaddis, Starting Out With Python (3e)

References:

1. Kenneth A. Lambert, Fundamentals of Python
2. Clinton W. Brownley, Foundations for Analytics with Python
3. James Payne, Beginning Python using Python 2.6 and Python 3
4. Charles Dierach, Introduction to Computer Science using Python
5. Paul Gries, Practical Programming: An Introduction to Computer Science using Python 3

Theory: 2 Hrs/Wk

SEC – IV

Total Credits: 2 Credits

UNIT - I

FILE SYSTEMS: Files, Directories, File System Implementation, File-SystemManagement and Optimization, Example File Systems.

INPUT/OUTPUT: Principles Of I/O Hardware, Principles Of I/O Software, I/O Software Layers, Disks, Clocks, User Interfaces: Keyboard, Mouse, Monitor, Thin Clients, Power Management.

UNIT- II

DEADLOCKS: Resources, Introduction to Deadlocks, Deadlock Detection and Recovery, Deadlock Avoidance, Deadlock Prevention.

SECURITY: The Security Environment, Operating Systems Security, Formal Models ofSecure Systems, Authentication, Insider Attacks, Malware, Defenses.

Text book:

1. A.S. Tanenbaum, and Herbert Bos, “Modern Operating Systems”, 4th Ed., Prentice- Hall of India, 2015.

References:

1. William Stallings, Operating Systems: Internals and Design Principles, 5th Ed., Prentice-Hall of India, 2006.
2. Gary Nutt, Operating Systems: A Modern Approach, 3rd Ed., Addison Wesley, 2004.
3. D.M. Dhamdhere, Operating Systems: A Concept Based Approach, 2nd Ed., Tata McGraw-Hill, 2007.

Information Technologies (Semester – IV)

Theory: 4 Hrs/Wk

General Elective

Total Credits: 4 Credits

UNIT – I

Computer Networks: Introduction, Connection Media, Data Transmission Mode, Data Multiplexing, Data Switching, Network Topologies, Types of Networks, Networking Devices, OSI Model.

The Internet: Internet Services, Types of Internet Connections, Internet Security.

Introduction to Emerging Computer Technologies: Distributed Networking, Peer-to-peer Computing, Grid Computing, Cloud Computing, Utility Computing, On-demand Computing, Wireless Network, Bluetooth, Artificial Intelligence.

UNIT – II

Email: Functions of Email, Browser, Web Browser, Internet Service Providers.

Introduction to Information Security: Need for Information Security, Threats to Information Systems, Information Assurance, Cyber Security.

Introduction to Application Security and Counter Measures: Introduction to Application Security, Data Security Considerations, Security Technologies, Security Threats, Security Threats to E-Commerce, E-Cash and Electronic Payment System.

UNIT – III

Introduction to Security Measures: Secure Information System Development Application Development Security Information Security Governance and Risk Management , Security Architecture and Design Security Issues in Hardware, Data Storage, and Downloadable Devices , Physical Security of IT Assets Backup Security Measures

UNIT – IV

Introduction to Security Policies and Cyber Laws: Need for an Information Security Policy, Information Security Standards – ISO, Introducing Various Security Policies and Their Review Process , Introduction to Indian Cyber Law Objective and Scope of the IT Act, 2000, Intellectual Property Issues, Overview of Intellectual-Property- Related Legislation in India, Patent, Copyright Law Related to Semiconductor Layout and Design , Software Licence

Text Books:

Dr. Surya Prakash T, Ritendra G, Praveen Kumar S, KLSI, Introduction to Information Security and Cyber Laws (Dreamtech Publication)

Theory: 4 Hrs/Wk (3 Credits)

Practical: 3 Hrs/Wk (1 Credit)

Total Credits: 3+1= 4 Credits

UNIT – I

Introducing PHP: What is PHP? Why use PHP? Evolution of PHP, Installing PHP, Other ways to run PHP, Creating your first script. PHP Language Basics – Using variables, Understanding Data Types, Operators and Expressions, Constants. Decisions and Loops – Making Decisions, Doing Repetitive Tasks with Looping, Mixing Decisions and Looping with HTML. Strings – Creating and Accessing Strings, Searching Strings, Replacing Text with Strings, Dealing with Upper and Lowercase, Formatting Strings. Arrays – Creating Arrays, Accessing Array Elements, Looping through Arrays with for-each,

UNIT – II

Creating Function, Reading Data in Web pages: setting up web pages to communicate with PHP, Handling Text Fields, Text Areas, Checkboxes, Radio Buttons, List Boxes, Password Controls, Image Maps, File Uploads, Buttons, and PHP Browser.

Object oriented programming: creating Classes and objects, setting access to properties and methods, constructors, destructors, Inheritance, overriding and overloading methods, auto loading classes.

UNIT – III

Advanced OOP: Static members and inheritance, Abstract classes, Interfaces, object iteration, comparing objects, class constants, final keyword, reflection.

File Handling: fopen, feof, fgets, closing a file, fgetc, file_get_contents, reading a file into an array with file, file_exists, filesize, fread, fscanf, parse_ini_file, stat, fseek, copy, unlink, fwrite, reading and writing binary files, appending a file, file_put_contents, locking files

UNIT – IV

Introducing Databases and SQL: Deciding How to Store Data, Understanding Relational Databases, Setting Up MySQL, A Quick Play with MySQL, Connecting MySQL from PHP. Retrieving Data from MySQL with PHP, Retrieving Data with SELECT, Creating a Member Record Viewer. Manipulating MySQL Data with PHP – Inserting, Updating, and Deleting Records, Building a Member Registration Application. Sessions, Cookies and FTP.

Text Books:

1. Steven Holzner, "PHP: The Complete Reference Paperback", McGraw Hill Education (India), 2007.
2. Timothy Boronczyk, Martin E. Psinas, "PHP and MYSQL (Create-Modify-Reuse)", Wiley India Private Limited, 2008.

PHP with MYYSQL – Lab

Practical: 3 Hrs/Wk (1 Credit)

1. Write a PHP script to display the Fibonacci sequence with HTML page.
2. Write a PHP script to create a chess board.
3. Write a PHP script using built-in string function like strstr(), strpos(), substr_count(), etc...
4. Write a PHP script to transform a string to uppercase, lowercase letters, make a string's first Character uppercase.
5. Write a PHP script to count number of elements in an array and display a range of array elements.
6. Write a PHP script using a function to display the entered string in reverse.
7. Write a PHP script to demonstrate inheritance.
8. Write a PHP script to demonstrate the object overloading with _get(), _set(), and _call().
9. Write a PHP script to demonstrate the method overloading and method overriding mechanisms.
10. Write a PHP script to demonstrate the use of final classes and final methods.
11. Write a PHP script to demonstrate the use of interfaces.
12. Write a PHP script using constructors and destructors.
13. Write a PHP application to handling HTML forms with PHP script.
14. Write a PHP script to create a file, write data into file and display the file's data.
15. Write a PHP script to check and change file permissions, copying, renaming and deleting files.
16. Write a PHP application for connecting to MySQL and reading data from database table.
17. Write a PHP application for inserting, updating, deleting records in the database table.
18. Develop a PHP application for student registration form.
19. Develop a PHP application for creating, updating, reading and deleting the Student records from MYSQL Database.

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
- External Vice-Voce is compulsory.

B.Sc. (Computer Science)
Model Question Paper (4 – Credits)

Time : 3 Hours

Max Marks : 80

PART -A

Answer any EIGHT questions in part –A

8 x 4 M = 32 Marks

UNIT- I

- 1
- 2
- 3

UNIT- II

- 4
- 5
- 6

UNIT- III

- 7
- 8
- 9

UNIT- IV

- 10
- 11
- 12

PART – B

Answer All Questions

12 x 4 = 48 Marks

UNIT- I

- 13
- OR**
- 14

UNIT- II

- 15
- OR**
- 16

UNIT- III

- 17
- OR**
- 18

UNIT- IV

- 19
- OR**
- 20

**B.Sc. (Computer Science)
Practical Question Paper (1 – Credit)**

Time: 3 Hours

Max Marks: 25

Answer any TWO questions (15 Marks)

UNIT – I 1 Program

UNIT – II 1 Program

UNIT – III 1 Program

UNIT – IV 1 Program

Viva – 5 Marks

Record – 5 Marks