# VTU Centre for Online Education (VTU-COE)

# Post Graduate Diploma in Big Data Analytics

# Scheme and Syllabus

# Postgraduate Diploma Programmes

| Program Outcomes | | |
|---|---|---|
| Sl No | Description | POs |
| 1 | Identify, critically analyze, formulate and develop computer applications | PO1 |
| 2 | Design a computing system to meet desired needs within realistic constraints such as safety, security and applicability | PO2 |
| 3 | Devise and conduct experiments interpret data and provide well informed conclusions. | PO3 |
| 4 | Understand the impact of system solutions in a contemporary, global, economical, environmental, and societal context for sustainable development | PO4 |
| 5 | Function professionally with ethical responsibility as an individual as well as in multidisciplinary teams with positive attitude | PO5 |
| 12 | Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change. | PO12 |

**Program Education Objectives (PEOs):**

**PEO 1:** Update their skills to keep up to the changing needs of society

**PEO2**: Demonstrate Managerial and practical skills and explore new areas like Data, Technology and knowledge discovery

**PEO3**: Security audit for full proof safe and secure system development.

# Scheme for Postgraduate Diploma Programmes

## SEMESTER-I

| Sl.No | Code | Course Name | Type of the Course | Credits |
|---|---|---|---|---|
| 1 | | Programming Using C | CORE | 5 |
| 2 | | Database Management System | CORE | 5 |
| 3 | | Computer Organization | CORE / SKILL | 5 |
| 4 | | DBMS Lab | SKILL | 2 |
| 6 | | C Programming Lab | SKILL | 3 |
| **Total Credits** | | | **20** | |

## SEMESTER-II

| Sl.No | Code | Course Name | Type of the Course | Credits |
|---|---|---|---|---|
| 1 | | Data Structures Using C++ | CORE | 5 |
| 2 | | Python Programming | CORE / SKILL | 5 |
| 3 | | System Software | CORE / SKILL | 5 |
| 4 | | Data Structures Lab | SKILL | 2 |
| 6 | | Python Lab | SKILL | 3 |
| **Total Credits** | | | **20** | |

**SEMESTER-I**

| Programming Using C | | Semester | I |
|---|---|---|---|
| Course Code | | CIE + SEE Marks | 30 + 70 = 100 |
| Credits | 3 | Exam Hours | 03 |
| Examination type (SEE) | Theory | | |

**Course objectives:**

**CLO1** : Explain user-defined data structures like arrays, structures,/unions and pointers in implementing solutions to problems
**CLO2**: Design and Develop Solutions to problems using modular programming constructs such as functions and procedures

**Module-1**

**Introduction to C**: Structure of C Program, Compiling and executing C programs, Variables, Constants, IO statements in C, Operators in C

**Module-2**

**Decision control and Looping statements**: Introduction to decision control, Conditional branching statements, iterative statements, nested loops, break and continue statements, goto statement.

**Module-3**

**Arrays**: Declaration of arrays, accessing and storing of values in array, Operations on arrays, 2-D arrays, operations on two-dimensional arrays, multidimensional arrays, applications of arrays
**Functions**: Introduction using functions, Function definition & declaration, function call, return statement, passing parameters to functions, Passing arrays to functions, scope of variables, storage classes, recursive functions.

**Module-4**

**Strings**: Introduction to strings, operations on strings, arrays of strings.

**Pointers**: Introduction to pointers, declaring pointer variables, Types of pointers, Passing arguments to functions using pointers.

**Module-5**

**Structure and Union**: Introduction, structures and functions, Unions, unions inside structures. **Files**: Introduction to files, Operation of Files.

**Course outcome (Course Skill Set)**

At the end of the course the student will be able to:

**CO1**: Illustrate the fundamental programming constructs of C programming language to solve problem.
**CO2**: Interpret the Use of functions and arrays in implementing solutions.
**CO3**: Demonstrate the use of structures, unions and pointers to solve problems.

**Suggested Learning Resources: Books**

1. Computer Fundamentals and Programming in C – Reema Thareja, 2nd Edition, Oxford University, 2017.
2. E. Balaguruswamy, Programming in ANSI C, 7th Edition, Tata McGraw-Hill
3. Brian W. Kernighan and Dennis M. Ritchie, The 'C' Programming Language, Prentice Hall of IndiaYashavanth Kanetkar, Let us C, Authentic Guide to C Programming Langauge, bpb publisher, 17th Edition, 2020
4. Yashavanth Kanetkar, Let us C, Authentic Guide to C Programming Langauge, bpb publisher, 17th Edition, 2020

**Web links and Video Lectures (e-Resources):**

- elearning.vtu.ac.in/econtent/courses/video/BS/15PCD23.html
- https://nptel.ac.in/courses/106/105/106105171/ MOOC courses can be adopted for more clarity in und

| Database Management System | | Semester | I |
|---|---|---|---|
| Course Code | | CIE +SEE Marks | 30 +70 =100 |
| Credits | 05 | Exam Hours | 3 |
| Examination type (SEE) | | Theory | |

**Course Objectives:**

**CLO 1**. Practice SQL programming through a variety of database problems.
**CLO 2**. Explore the use of concurrency and transactions in database.
**CLO 3**. Build database applications for real world problems.

**MODULE 1**

Introduction: Characteristics of Database approach, Actors on the Scene, Workers behind the scene, Advantages of using DBMS approach, Data models, schemas and instances, Three -schema architecture and data independence, Database languages and interfaces, the database system environment, Centralized and client -server architectures, Classification of Database Management systems, Entity-Relationship Model: Conceptual Database using high level conceptual data models for Database Design, A Sample Database Application, Entity types, Entity sets Attributes and Keys Relationship types, Relationship Sets, Roles and Structural Constraints Weak Entity Types.

**MODULE 2**

Relational Model Relational Model and Relational Algebra: Relational Model Concepts, Relational Model Constraints and Relational Database Schema Update Operations, Transactions and Dealing with Constraint violations, Unary Relational operations, Relational Algebra Operations from Set Theory, Binary Relational Operations, JOIN and DIVISION, Additional Relational Operations, Examples of Queries in Relational Algebra Relational Database Design Using ER-to Relational Mapping .

**MODULE 3**

Introduction to SQL: Overview of the SQL Query Language, SQL Data Definition, Basic structure of SQL Queries, Additional Basic Operations, Null values, Aggregate Functions, nested Sub queries, Modification of the Database, Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization.

| MODULE 4 |
| --- |
| Database Design: Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definitions of 2nd and 3rd Normal Forms, Boyce Codd Normal Forms, Stored Procedures and functions, Triggers. |
| **MODULE 5** |
| Transaction Management: Transaction Concept, A Simple Transaction Model, Transaction Atomicity and Durability, Serializability, Transaction Isolation and Atomicity, Transaction Isolation Levels, Implementation of Isolation Levels. Concurrency Control: Lock Based Protocols, Deadlock Handling. Recovery System: Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm. |
| **Course outcome (Course Skill Set):**<br>At the end of the course the student will be able to :<br><br>**CO1:** Figure out the concepts of database objects, enforce integrity constraints on a database using RDBMS.<br>**CO2**: Demonstrate Structured Query Language (SQL) for database manipulation and also the basic of query evaluation.<br>**CO3**: Develop application to interact with databases, relational algebra expression,<br>**CO4**: Construct an application using tuple and domain relation expression from queries.<br><br>**Recommended Text and Reference Books:**<br>1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.<br>2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill.<br>3. Abraham Silberschatz, Henry F. Korth and S. Sudarshan‟s Database System Concepts 9th EditionTata Mcgraw Hill Education Private Limited-2013<br>4. Introduction to Database Management System ,Satinder bal Gupta,Aditiya Mittal,2nd Edition,An imprint of Laxmi publications Private Limited-2017 |

| Computer Organization | | Semester | I |
|---|---|---|---|
| Course Code | | CIE +SEE Marks | 30 +70 =100 |
| Credits | 05 | Exam Hours | 3 |
| Examination type (SEE) | | Theory | |

**Course Objectives:**

**CLO 1**. Practice SQL programming through a variety of database problems.
**CLO 2**. Explore the use of concurrency and transactions in database.
**CLO 3**. Build database applications for real world problems.

**MODULE 1**

**Basic Structure of Computers:**
Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement.

**Machine Instructions and Programs:**

Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions

Text book 1: Chapter1 – 1.3, 1.4, 1.6 (1.6.1-1.6.4, 1.6.7), Chapter2 – 2.2 to 2.10

**MODULE 2**

**Input/Output Organization:**
Accessing I/O Devices, Interrupts – Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB.
Text book 1: Chapter4 – 4.1, 4.2, 4.4, 4.5, 4.6, 4.7

**MODULE 3**

**Memory System:**

Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations.
Text book 1: Chapter5 – 5.1 to 5.4, 5.5 (5.5.1, 5.5.2), 5.6

## MODULE 4

**Arithmetic:**

Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication, Integer Division.

Text book 1: Chapter2-2.1, Chapter6 – 6.1 to 6.6

## MODULE 5

**Basic Processing Unit:**

Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro programmed Control.

**Pipelining:**

Basic concepts of pipelining,

Text book 1: Chapter7, Chapter8 – 8.1

**Course outcome (Course Skill Set):**

At the end of the course the student will be able to :

**CO1:** Explain the basic organization of a computer system.
**CO2** : Demonstrate functioning of different sub systems, such as processor, Input/output,and memory.
**CO3**: Illustrate hardwired control and micro programmed control, pipelining, embedded and other computing systems.
**CO4**: Design and analyze simple arithmetic and logical units.

**Recommended Text and Reference Books:**

1. William Stallings: Computer Organization & Architecture, 9th Edition, Pearson, 2015.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, 5th Edition, Tata McGraw Hill, 2002. (Listed topics only from Chapters 1, 2, 4, 5, 6, 7, 8, 9 and12)
3. M. Morris Mano, Computer System Architecture, PHI, 3rd Edition
4. Essentials of Computer Organization and Architecture, 5th Edition by Linda Null, Julia Lobur

| Database Management Lab | | Semester | I |
|---|---|---|---|
| Course Code | | CIE + SIE Marks | 30 +70 = 100 |
| Credits | 02 | Exam Hours | 03 |
| Examination type (SEE) | Practical | | |

**Course objectives:**

**CLO1**: Create SQL queries for the small projects.

**CLO2**: Create database objects that include tables, constraints, indexes, and sequences.

1. Students should be allowed to choose appropriate DBMS software, install it, configure it and start working on it. Create sample tables, execute some queries, use SQLPLUS features, Use PL/SQL features like cursors on sample database. Students should be permitted to practice appropriate User interface creation tool and Report generation tool.

2. A college consists of number of employees working in different departments. In this context, create two tables' employee and department. Employee consists of columns empno, empname, basic, hra, da, deductions, gross, net, date-of-birth. The calculation of hra,da are as per the rules of the college. 1. Create tables department and employee with required constraints. 2. Initially only the few columns (essential) are to be added. Add the remaining columns separately by using appropriate SQL command 3. Basic column should not be null 4. Add constraint that basic should not be less than 5000. 5. Calculate hra,da,gross and net by using PL/SQL program.

3. Students may be divided into batches and the following experiments may be given to them to better understand the DBMS concepts. Students should gather the required information, draw ER diagrams, map them to tables, normalize, create tables, triggers, procedures, execute queries, create user interfaces, and generate reports.

- Student information system
- KSRTC reservation system
- Hostel management
- Library management
- Indian Railways reservation

**Course outcomes (Course Skill Set):**

At the end of the course the student will be able to:

**CO1**: Design entity-relationship diagrams to solve given database applications.

**CO2**: Implement a database schema for a given problem.

**CO3**: Formulate SQL queries in Oracle for the given problem.

**CO4**: Design and Develop suitable database and verify for its appropriate normalization for any given problem.

| C Programming Lab | | Semester | I |
|---|---|---|---|
| Course Code | | CIE + SEE Marks | 30 +70 =100 |
| Credits | 03 | Exam Hours | 03 |
| Examination type (SEE) | Practical | | |

**Course objectives:**

**CLO1** : Exploring an programs using constructs of C programming language
**CLO2**: Demonstrate the use of IDE, C Compiler, and identify and rectify the syntax and syntactic errors during programming.
**CLO3**: Learn to Reporting the observations and debug the program.

**Laboratory Experiments:**

Implement the following programs with WINDOWS / LINUX platform using appropriate C compiler

1. Implement a C program that takes three coefficients (a, b, and c) of a Quadratic equation ($ax^2+bx+c=0$) as input and compute all possible roots, output the roots with appropriate messages.
2. Write a C program to simulate a Simple Calculator using Switch case construct.
3. Develop a C Program to check whether a given number is PALINDROME or NOT. Ex: Num: 1221, Reverse: 1221, It is a Palindrome
4. Design and develop a C program to read a year as an input and find whether it is leap year or not.
5. Develop a C Program to search a Name in a list of names using Binary searching Technique (Use strcmp built-in function).
6. Write a C program that reads N integer numbers and arrange them in ascending order using Bubble Sort.
7. Develop, implement and execute a C program that reads two matrices A (m x n ) and B (p x q ) and Compute product of matrices A and B. Print both the input matrices and resultant matrix with suitable headings and output should be in matrix format only. Program must check the compatibility of orders of the matrices for multiplication. Report appropriate message in case of incompatibility.
8. Design and develop a C function isprime(num) that accepts an integer argument and returns 1 if the argument is prime, a 0 otherwise. Write a C program that invokes this function to generate prime numbers between the given range.
9. write a recursive C function to find the factorial of a number, n!, defined by fact(n)=1, if n=0. Otherwise fact(n)=n*fact(n-1). Using this function, write a C program to compute the binomial coefficient nCr . Tabulate the results for different values of n and r with suitable messages.
10. Write a C program to copy the contents of one file to another.

**11.** Write a C program that uses functions and structures to perform the following operations:

- **a.** Reading a complex number
- **b.** Displaying a complex number
- **c.** Addition of two complex numbers
- **d.** Multiplication of two complex numbers

Display the appropriate output.

12. Write a Program in c to swap two number using pointer.

**Course outcome (Course Skill Set)**

At the end of the course the student will be able to:

**CO1**: Make use of IDE for programming, identify and correct the syntax and syntactic errors using various programming constructs.

**CO2**: Demonstrate use of functions, recursive functions, arrays, strings, structures and pointers in problem solving.

**CO3**: Design and development of C programs to implement different searching and sorting techniques.

**Suggested Learning Resources:**

**Books**

1. Computer Fundamentals and Programming in C – Reema Thareja, 2nd Edition, Oxford Univeristy, 2017
2. E. Balaguruswamy, Programming in ANSI C, 7th Edition, Tata McGraw-Hill
3. Brian W. Kernighan and Dennis M. Ritchie, The 'C' Programming Language, Prentice Hall of India
4. Yashavanth Kanetkar, Let us C, Authentic Guide to C Programming Langauge, bpb publisher, 17th Edition, 2020

**Web links and Video Lectures (e-Resources):**

- elearning.vtu.ac.in/econtent/courses/video/BS/14CPL16.html
- https://nptel.ac.in/courses/106/105/106105171/

| Data Structures Using C++ | | Semester | II |
|---|---|---|---|
| Course Code | | CIE + SIE Marks | 30 + 70 =100 |
| Credits | 05 | Exam Hours | 03 |
| Examination type (SEE) | Theory | | |

**Course Learning objectives:**

**CLO1**: Explore step by step and develop algorithms to solve real world problems.

**CLO2**: Evaluate the Expressions like postfix, prefix conversions.

**CLO3**: Implementing various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs.

**CLO4**: Define various searching & sorting techniques.

| Module-1 | |
|---|---|
| **INTRODUCTION:** C++ features, Data types, Variables, Operators, Expressions, Control structures, classes and Objects, Functions and parameters, function overloading, Recursion, Constructors, Destructors and Operator overloading, Inheritance, Polymorphism, Programming examples. | |
| **Teaching-Learning Process** | Chalk and talk method / PowerPoint Presentation |

| Module-2 | |
|---|---|
| **ARRAYS AND MATRICES:** Arrays, Matrices, Special matrices, Sparse matrices. **POINTERS:** Pointers, Dynamic memory allocation **LINEAR LISTS:** Data objects and structures, Introduction to Linear and Non-Linear data structures, Linear list data structures, Array Representation, Vector Representation, Singly Linked lists and chains. | |
| **Teaching-Learning Process** | Chalk and talk method / PowerPoint Presentation |

| Module-3 | |
|---|---|
| **STACKS:** The abstract data types, Array Representation, Linked Representation, Applications – Parsing and Evaluation of arithmetic expressions, Parenthesis Matching & Towers of Hanoi. | |
| **Teaching-Learning Process** | Chalk and talk method / PowerPoint Presentation |

| Module-4 | |
|---|---|
| **QUEUES:** The abstract data types, Array Representation, Linked Representation, Applications-Railroad car arrangement, Priority Queues HASHING: Dictionaries, Linear representation, Hash table representation. | |
| **Teaching-Learning Process** | Chalk and talk method / PowerPoint Presentation |
| **Module-5** | |
| **TREES:** Binary trees, Properties and representation of binary trees, Common binary tree operations, Binary tree traversal the ADT binary tree, ADT binary tree and the class linked binary tree. Binary search trees operations and implementation. Heaps, Applications—Heap Sorting | |
| **Teaching Learning Process** | Chalk and talk method / PowerPoint Presentation |

**Course outcome (Course Skill Set):**

At the end of the course the student will be able to:

**CO1:** Distinguish between procedures and object-oriented programming. Read more at:

**CO2:** Apply advanced data structure strategies for exploring complex data structures Read more

**CO3:** Implement data structure algorithms through C++.

**CO4** : Incorporate data structures into the applications such as binary search trees,

**Suggested Learning Resources: Reference books:**

1. Data structures, Algorithms and Applications in C++,
2. S.Sahni, University Press (India) Pvt.Ltd, 2nd edition, Universities Press, Pvt. Ltd.
3. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.
4. Data structures and Algorithms in C++, Michael T.Goodrich, R.Tamassia and .Mount, Wiley student edition, John Wiley and Sons.

| Object Oriented Programming Using Python | | Semester | II |
|---|---|---|---|
| Course Code | | CIE + SIE Marks | 30 + 70 =100 |
| Credits | 5 | Exam Hours | 3 |
| Examination type (SEE) | Theory | | |

**Course objectives:**

**CLO1**: Study the importance of Object Oriented Programming
**CLO2**: Explore the Object Oriented Programming concepts
**CLO3**: Explain the concept of Polymorphism, Inheritance
**CLO4**: Understand the creation of modules ,packages and organization of modules and packages

**Teaching-Learning Process (General Instructions)**

Programming Exercises and mini project works.

**Module-1**

Python Basic Concepts and Programming

Parts of Python Programming Language, Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, The type( ) Function and Is Operator, Control Flow Statements, The if Decision Control Flow Statement, The if…else Decision Control Flow Statement, The if…elif…else Decision Control Statement, Nested if Statement, The while Loop, The for Loop, The continue and break Statements, Built-In Functions, Commonly Used Modules, Function Definition and Calling the Function, The return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

**Module-2**

Python Collection Objects, Strings- Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings, Lists-Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods. Sets, Tuples and Dictionaries. Files: reading and writing files

**Module-3**

Object-oriented Design :Introducing object-oriented ,Objects and classes, Specifying attributes and behaviours : Data describes objects , Behaviours are actions ,Hiding details and creating the public interface: Composition, Inheritance: Inheritance provides abstraction, Multiple inheritance Objects in Python : Creating Python classes , Adding attributes, Making a function work: passing arguments, Initializing the object, self argument

**Module-4**

Modules and packages: Organizing the modules, Absolute imports, Relative imports, Organizing module contents: Access control, Third-party libraries, Basic inheritance, Extending built-ins, Overriding and super, Multiple inheritance, The diamond problem, Different sets of arguments,

**Module-5**

Polymorphism , Abstract base classes, Using an abstract base class , Creating an abstract base class Exceptions: Raising exceptions, The effects of an exception , Handling exceptions, The exception hierarchy, Defining our own exceptions

**Course outcome (Course Skill Set)**

At the end of the course the student will be able to:

**CO1**: Demonstrate proficiency in handling loops and creation of functions

**CO2**: Illustrate the methods to create and manipulate lists, tuples and dictionaries .

CO3: Design and Develop programs for string processing and file organization.

**CO4**: Interpret the concept of OOP as used in Python

**Suggested Learning Resources:**

**Books**

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd edition,Updated for Python 3, Shroff/O'Reilly Publishers, 2016
2. Python 3 Object Oriented Programming, 2nd Edition, Unleash the power of Python 3 Objects by Dusty Phillips , PACKT Publishing.
3. Python Object–Oriented Programming :Build robust and maintainable Object-oriented python applications and libraries, Steven F. Lott, Dusty Philips,4th Edition, Packt Publishing Limited; 2021
4. Python the complete reference ,Martin C. Brown,4th Edition, McGraw Hill Education ,2018

**Web links and Video Lectures (e-Resources):**

- http://greenteapress.com/wp/thinkpython/

| Software Engineering with Agile Methodologies | | Semester | II |
|---|---|---|---|
| Course Code | | CIE + SIE Marks | 30 + 70 = 100 |
| Credits | 05 | Exam Hours | 03 |
| Examination type (SEE) | | Theory | |

**Course Learning objectives:**

**CLO1**: Outline software engineering principles and activities involved in building large software programs.
**CLO2**: Identify ethical and professional issues and explain why they are of concern to software engineers.
**CLO3**: Explain the fundamentals of object oriented concepts.
**CLO4**: Describe the process of requirements gathering, requirements classification, requirements specification and requirements validation.
**CLO5**: Differentiate system models, use UML diagrams and apply design patterns.
**CLO6**: Discuss the distinctions between validation testing and defect testing.

**MODULE 1**

Introduction to System Software, Machine Architecture of SIC and SIC/XE. Assemblers: Basic assembler functions, machine dependent assembler features, machine independent assembler features, assembler design options. Basic Loader Functions

Text book 1: Chapter 1: 1.1,1.2,1.3.1,1.3.2, Chapter2 : 2.1 to 2.4, Chapter 3 ,3.1

**MODULE 2**

**Introduction**: Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building compiler, Applications of compiler technology. Lexical Analysis: The role of lexical analyzer, Input buffering, Specifications of token, recognition of tokens.

Text book 2:Chapter 1 1.1-1.5 Chapter 3: 3.1 – 3.4

**MODULE 3**

Syntax Analysis: Introduction, Context Free Grammars, Writing a grammar, Top Down Parsers, Bottom-Up Parsers

Text book 2: Chapter 4 4.1, 4.2 4.3 4.4 4.5

| MODULE 4 |
| --- |
| Lex and Yacc –The Simplest Lex Program, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program,<br>Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity. Text book 3: Chapter 1,2 and 3. |
| **MODULE 5** |
| Syntax Directed Translation, Intermediate code generation, Code generation<br>Text book 2: Chapter 5.1, 5.2, 5.3, 6.1, 6.2, 8.1, 8.2 |
| **Course outcome (Course Skill Set)**<br>**At the end of the course the student will be able to :**<br>**CO1:** Explain system software<br>**CO2** : Design and develop lexical analyzers, parsers and code generators<br>**CO3**: Utilize lex and yacc tools for implementing different concepts of system software<br>**CO4**: Learn the basic design and working of compilers.<br><br>**Suggested Learning Resources:**<br>**Reference books:**<br>1. J. Nithyashri, "System Software", 2nd Edition, Tata McGraw Hill, 2010.<br>2. System Software by Leland. L. Beck, D Manjula, 3rd edition, 2012<br>3. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman , Compilers-Principles, Techniques and Tools, Pearson, 2nd edition, 2007<br>4. Doug Brown, John Levine, Tony Mason, lex & yacc, O'Reilly Media, October 2012. |

| Data Structures Laboratory | | Semester | II |
|---|---|---|---|
| Course Code | | CIE + SIE Marks | 30 + 70 = 100 |
| Credits | 2 | Exam Hours | 03 |
| Examination type (SEE) | Practical | | |

**Course objectives:**
**CLO1**: Explain the Evaluation of Expressions like postfix, prefix conversions.
**CLO2**: Implementing various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs.

| Sl. No | Experiments |
|---|---|
| 1 | Implement a Program in C++ for converting an Infix Expression to Postfix Expression. |
| 2 | Design, develop, and execute a program in C++ to evaluate a valid postfix expression using stack. Assume that the postfix expression is read as a single line consisting of non-negative single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), * (multiply) and / (divide). |
| 3 | Design, develop, and execute a program in C++ to simulate the working of a queue of integers using an array. Provide the following operations: a. Insert b. Delete c. Display |
| 4 | Write a C++ program to simulate the working of a singly linked list providing the following operations: a. Display & Insert b. Delete from the beginning/end c. Delete a given element |
| 5 | Write a C++ program to Implement the following searching techniques a. Linear Search b. Binary Search. |
| 6 | Write a C++ program to implement the following sorting algorithms using user defined functions: a. Bubble sort (Ascending order) b. Selection sort (Descending order). |
| 7 | Write a C++ program to implement the Binary Search Tree operations. |
| 8 | Write a C++ program to demonstrate the Binary Tree Traversals - Inorder, postorder, preorder |

**Course outcomes (Course Skill Set):**

At the end of the course the student will be able to:

**CO1**: Design and Develop Data structure techniques for evaluating the given expression.
**CO2**: Demonstrate various sorting / searching techniques and validate input/output for the given problem.
**CO3**: Design data structures to show the operations on Stacks, Queues, Circular Queues, Linked Lists, and Trees.
**CO4**: Implement the suitable algorithm to find whether the given graph is connected or not and illustrate the performance of the technique implemented.

| Python Lab | | Semester | II |
|---|---|---|---|
| Course Code | | CIE + SIE Marks | 30 + 70 =100 |
| Credits | 03 | Exam Hours | 03 |
| Examination type (SEE) | | Practical | |

**Course Learning Objectives:**

**CLO1**: To be able to introduce core programming basics and program design with functions using Python programming language.

**CLO2**: To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.

**CLO3**: To understand the high-performance programs designed to strengthen the practical expertise.

| |
|---|
| 1. Write a program to sum all the elements from n1 to n2 where n1 and n2 are positive integers |
| 2. Input an array of n numbers and find separately the sum of positive numbers and negative numbers. |
| 3. Write a program to search an element using linear search |
| 4. Write a program to search an element using binary search. |
| 5. Write a program to simulate stack. |
| 6. Using a stack evaluate an arithmetic expression. |
| 7. Write a program to multiply two matrices |
| 8. Write a program to find the roots of a quadratic equation |
| 9. Write a program to Insert a number in a sorted array. |
| 10. Write a Python Program to check whether the given string is palindrome or not using built in string manipulation methods. |
| 11. Write a Python Program to read a word and prints the number of letters, vowels and percentage of vowels in the word using dictionary |
| 12. Write a Python Program to check a given sentence is a pangram or not using function/Module. |

**Course Outcomes:** At the end of the course the student will be able to:

**CO1**: Demonstrate proficiency in handling loops and creation of functions.

**CO2**: Illustrate the methods to create and manipulate lists, tuples and dictionaries in Python Programme.

**CO3**: Design and Develop programs for string processing and file organization and use the concept of OOP as used in Python.