ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

I SEMESTER	**COMPUTER SCIENCE**	TIME:4HRS/WEEK
CS 1602(3)	**PROBLEM SOLVING IN 'C'**	MAX.MARKS:100

w.e.f.20-21 admitted batch-"20AH" **SYLLABUS**

OBJECTIVES: This course aims to provide exposure to problem-solving through programming. It introduces the concepts of the C Programming language.

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, a student will be able to:

1. Understand the evolution and functionality of a Digital Computer.
2. Apply logical skills to analyse a given problem
3. Develop an algorithm for solving a given problem.
4. Understand 'C' language constructs like Iterative statements, Array processing, Pointers, etc.
5. Apply 'C' language constructs to the algorithms to write a 'C' language program.

UNIT – I: General Fundamentals: Introduction to computers, Block diagram of a computer, characteristics and limitations of computers, applications of computers, types of computers, computer generations.

Introduction to Algorithms and Programming Languages: Algorithm – Key features of Algorithms, Flow Charts, Programming Languages – Generations of Programming Languages – Structured Programming Language- Design and Implementation of Correct, Efficient and Maintainable Programs.

UNIT – II: Introduction to C: Introduction – Structure of C Program – Writing the first C Program –File used in C Program – Compiling and Executing C Programs – Using Comments – Keywords – Identifiers – Basic Data Types in C – Variables – Constants – I/O Statements in C- Operators in C- Programming Examples.

Decision Control and Looping Statements : Introduction to Decision Control Statements– Conditional Branching Statements – Iterative Statements – Nested Loops – Break and Continue Statement – Goto Statement.

UNIT – III: Arrays : Introduction – Declaration of Arrays – Accessing elements of the Array – Storing Values in Array– Operations on Arrays – one dimensional, two dimensional and multidimensional arrays, character handling and strings.

UNIT – IV: Functions: Introduction – using functions – Function declaration/ prototype – Function definition function call – return statement – Passing parameters – Scope of variables –Storage Classes – Recursive functions. Structure, Union, and Enumerated Data Types: Introduction – Nested Structures – Arrays of Structures – Structures and Functions– Union – Arrays of Unions Variables – Unions inside Structures – Enumerated Data Types.

UNIT – V: Pointers: Understanding Computer Memory – Introduction to Pointers – declaring Pointer Variables – Pointer Expressions and Pointer Arithmetic – Null Pointers – Passing Arguments to Functions using Pointer – Pointer and Arrays – Memory Allocation in C Programs –Memory Usage – Dynamic Memory Allocation – Drawbacks of Pointers.

Files: Introduction to Files – Using Files in C – Reading Data from Files – Writing Data to Files – Detecting the End-of-file – Error Handling during File Operations – Accepting Command Line Arguments.

REFERENCE BOOKS :

1. E Balagurusamy – Programming in ANSI C – Tata McGraw-Hill publications.
2. Brain W Kernighan and Dennis M Ritchie – The 'C' Programming language" –Pearson publications.
3. Ashok N Kamthane: Programming with ANSI and Turbo C, Pearson Edition Publications.
4. YashavantKanetkar – Let Us 'C' – BPB Publications.

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

I SEMESTER **COMPUTER SCIENCE** TIME:2HRS/WEEK

CS 1652 (2) **PROBLEM SOLVING IN 'C' LAB** MAX.MARKS:50

w.e.f. 20-21 admitted batch-"20AH" **PRACTICAL SYLLABUS – IA**

1. Write a program to check whether the given number is Armstrong or not.
2. Write a program to find the sum of individual digits of a positive integer.
3. Write a program to generate the first n terms of the Fibonacci sequence.
4. Write a program to find both the largest and smallest number in a list of integer values
5. Write a program to demonstrate reflection of parameters in swapping of two integer values using Call by Value & Call by Address
6. Write a program that uses functions to add two matrices.
7. Write a program to calculate factorial of given integer value using recursive functions
8. Write a program for multiplication of two N X N matrices.
9. Write a program to perform various string operations.
10. Write a program to search an element in a given list of values.
11. Write a program to sort a given list of integers in ascending order.
12. Write a program to calculate the salaries of all employees using Employee (ID,Name, Designation, Basic Pay, DA, HRA, Gross Salary, Deduction, Net Salary)structure.
    a. DA is 30 % of Basic Pay
    b. HRA is 15% of Basic Pay
    c. Deduction is 10% of (Basic Pay + DA)
    d. Gross Salary = Basic Pay + DA + HRA
    e. Net Salary = Gross Salary - Deduction
13. Write a program to illustrate pointer arithmetic.
14. Write a program to read the data character by character from a file.
15. Write a program to createBook (ISBN,Title, Author, Price, Pages,Publisher)structure and store book details in a file and perform the following operations
    a. Add book details b. Search a book details for a given ISBN and display book details, if available
    c. Update a book details using ISBN
    d. Delete book details for a given ISBN and display list of remaining Books

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

II SEMESTER **COMPUTER SCIENCE** TIME:4HRS/WEEK

CS 2602(3) **DATA STRUCTURES USING 'C'** MAX.MARKS:100

20-21 admitted batch-"20AH" **SYLLABUS**

COURSE OBJECTIVES: To Introduce the fundamental concept of data structures and to emphasize the importance of various data structures in developing and implementing efficient algorithms.

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, a student will be able to:

1. Understand available Data Structures for data storage and processing.
2. Comprehend Data Structure and their real-time applications - Stack, Queue, Linked List, Trees and Graph
3. Choose a suitable Data Structures for an application.
4. Develop ability to implement different Sorting and Search methods
5. Have knowledge on Data Structures basic operations like insert, delete, search, update and traversal
6. Design and develop programs using various data structures
7. Implement the applications of algorithms for sorting, pattern matching etc

UNIT – I: Introduction to Data Structures: Introduction to the Theory of Data Structures, Data Representation, Abstract Data Types, Data Types, Primitive Data Types, Data Structure and Structured Type, Atomic Type, Difference between Abstract Data Types, Data Types, and Data Structures, Refinement Stages.

Principles of Programming and Analysis of Algorithms: Software Engineering, Program Design, Algorithms, Different Approaches to Designing an Algorithm, Complexity, Big 'O' Notation, Algorithm Analysis, Structured Approach to Programming, Recursion, Tips and Techniques for Writing Programs in 'C'.

UNIT – II: Arrays: Introduction to Linear and Non- Linear Data Structures, One- Dimensional Arrays, Array Operations, Two- Dimensional arrays, Multidimensional Arrays, Pointers and Arrays, an Overview of Pointers

Linked Lists: Introduction to Lists and Linked Lists, Dynamic Memory Allocation, Basic Linked List Operations, Doubly Linked List, Circular Linked List, Atomic Linked List,Linked List in Arrays, Linked List versus Arrays.

UNIT – III: Stacks: Introduction to Stacks, Stack as an Abstract Data Type, Representation of Stacks through Arrays, Representation of Stacks through Linked Lists, Applications of Stacks, Stacks and Recursion

Queues: Introduction, Queue as an Abstract data Type, Representation of Queues, Circular Queues, Double Ended Queues- Deques, Priority Queues, Application of Queues.

UNIT – IV: Binary Trees: Introduction to Non- Linear Data Structures, Introduction Binary Trees, Types of Trees, Basic Definition of Binary Trees, Properties of Binary Trees, Representation of Binary Trees, Operations on a Binary Search Tree, Binary Tree Traversal, Counting Number of Binary Trees, Applications of Binary Tree.

UNIT – V: Searching and sorting: Sorting – An Introduction, Bubble Sort, Insertion Sort, Merge Sort, Searching – An Introduction, Linear or Sequential Search, Binary Search, Indexed Sequential Search.

Graphs: Introduction to Graphs, Terms Associated with Graphs, Sequential Representation of Graphs, Linked Representation of Graphs, Traversal of Graphs, Spanning Trees, Shortest Path, Application of Graphs.

REFERENCE BOOKS :

1. "Data Structures using C", ISRD group Second Edition, TMH
2. "Data Structures through C", YashavantKanetkar, BPB Publications
3. "Data Structures Using C" Balagurusamy E. TMH.

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM
II SEMESTER **COMPUTER SCIENCE** TIME:3HRS/WEEK
CS 2652 (2) **DATA STRUCTURES USING 'C' LAB** MAX.MARKS:50
20-21 admitted batch-"20AH" **PRACTICAL SYLLABUS**

1. Write a program to read 'N' numbers of elements into an array and also perform the following operation on an array
   a. Add an element at the begging of an array
   b. Insert an element at given index of array
   c. Update a element using a values and index
   d. Delete an existing element
2. Write a program using stacks to convert a given
   a. postfix expression to prefix
   b. prefix expression to postfix
   c. infix expression to postfix
3. Write Programs to implement the Stack operations using an array.
4. Write Programs to implement the Stack operations using Linked List.
5. Write Programs to implement the Queue operations using an array.
6. Write Programs to implement the Queue operations using Linked List.
7. Write a program for arithmetic expression evaluation.
8. Write a program for Binary Search Tree Traversals.
9. Write a program to implement dequeue using a doubly linked list.
10. Write a program to search an item in a given list using the following Searching Algorithms
    a. Linear Search
    b. Binary Search.
11. Write a program for implementation of the following Sorting Algorithms
    a. Bubble Sort
    b. Insertion Sort
    c. Quick Sort
12. Write a program for polynomial addition using single linked list.
13. Write a program to find the shortest path between a given Source Node and Destination Node in a given graph using Dijkstrar's algorithm.
14. Write a program to implement Depth First Search graph traversals algorithm.
15. Write a program to implement Breadth First Search graph traversals algorithm.

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM
III SEMESTER **COMPUTER SCIENCE** TIME:4HRS/WEEK
CS 3602 (3) **DATABASE MANAGEMENT SYSTEMS** MAX.MARKS:100
w.e.f. 20-21 admitted batch-"20AH" **SYLLABUS**

COURSE OBJECTIVES: To enable the students to:

- Understand the different issues involved in the design and implementation of a database system.
- To understand and use data manipulation language to query, update, and manage a database.
- To introduce the concepts of transactions and transaction processing.

COURSE OUTCOMES:

Upon successful completion of the course, a student will be able to:

- Develop and design database application and therefore enhance entrepreneurship skills.
- Design entity relationship and convert entity relationship diagrams into RDBMS and formulate SQL queries on the respect data.
- Design and implement a Database Schema for a given Problem-domain.
- Apply Normalization Techniques on given Database Design to avoid Anomalies.
- Understand various transaction processing and concurrency control mechanisms.

UNIT I : OVERVIEW OF DATABASE MANAGEMENT SYSTEM: Introduction to data, information, database, database management systems, file-based system, Drawbacks of file-Based System, database approach, Classification of Database Management Systems, advantages of database approach, Various Data Models, Components of Database Management System, three schema architecture of data base, costs and risks of database approach.

UNIT II : ENTITY-RELATIONSHIP MODEL: Introduction, the building blocks of an entity relationship diagram, classification of entity sets, attribute classification, relationship degree, relationship classification, reducing ER diagram to tables, enhanced entity-relationship model (EER model), generalization and specialization, IS A relationship and attribute inheritance, multiple inheritance, constraints on specialization and generalization, advantages of ER modelling.

UNIT III : RELATIONAL MODEL: Introduction, CODD Rules, relational data model, concept of key, relational integrity, relational algebra, relational algebra operations, advantages of relational algebra, limitations of relational algebra, relational calculus, tuple relational calculus, domain relational Calculus (DRC), Functional dependencies and normal forms upto 3rd normal form.

UNIT IV : STRUCTURED QUERY LANGUAGE: Introduction, History of SQL Standard, Commands in SQL, Data Types in SQL, Data Definition Language, Selection Operation, Projection Operation, Aggregate functions, Data Manipulation Language, Table Modification Commands, Join Operation, Set Operations, View, Sub Query.

UNIT V : PL/SQL: Introduction, Shortcomings of SQL, Structure of PL/SQL, PL/SQL Language Elements, Data Types, Operators Precedence, Control Structure, Steps to Create a PL/SQL, Program, Iterative Control, Procedure, Function, Database Triggers, Types of Triggers.

PRESCRIBED TEXT BOOKS:

- Database System Concepts by Abraham Silberschatz, Henry Korth, and S. Sudarshan, McGrawhill ,6e.
- Database Management Systems by Raghu Ramakrishnan, McGrawhill ,3e

REFERENCE BOOKS:

- Principles of Database Systems by J. D. Ullman
- Fundamentals of Database Systems by R. Elmasri and S. Navathe
- SQL: The Ultimate Beginners Guide by Steve Tale.

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM
III SEMESTER **COMPUTER SCIENCE** TIME:2HRS/WEEK
CS 3652 (2) **DATABASE MANAGEMENT SYSTEMS LAB** MAX.MARKS:50
w.e.f.20-21 admitted batch-"20AH" **PRACTICAL SYLLABUS**

OBJECTIVES :

To enable the students to:

- Describe the basics of SQL and construct queries using SQL.
- Know query languages associated with relational models

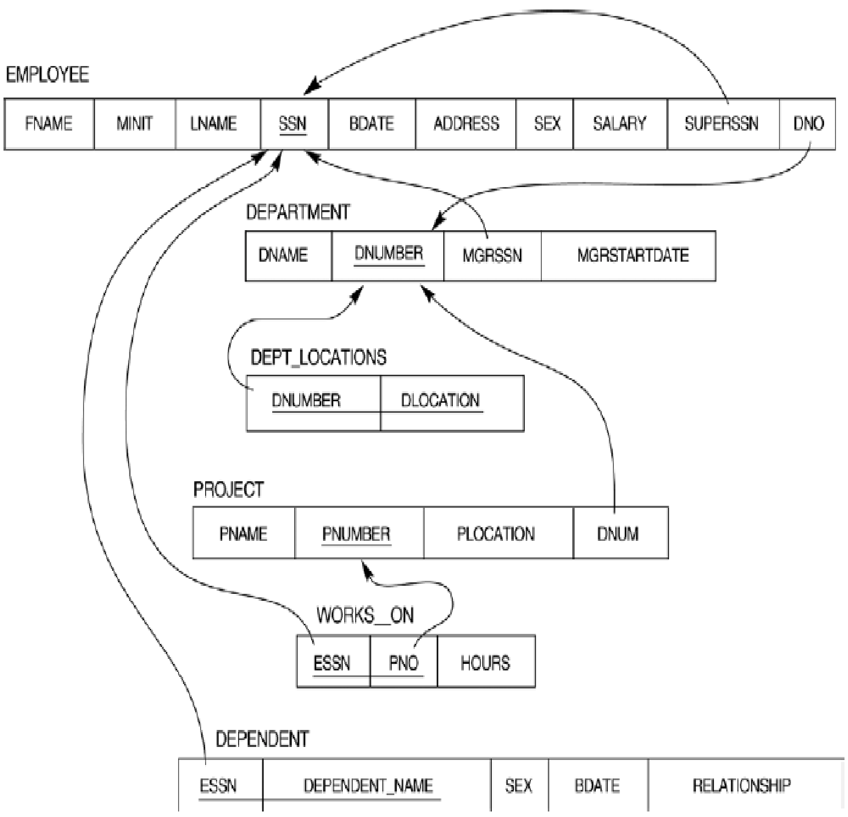
COURSE OUTCOMES:

After Completion of this course the student would be able to:

- Design and implement a database schema for a given problem.
- Design queries using SQL.
- Apply PL/SQL for processing database.

1. Draw ER diagram for hospital administration
2. Creation of college database and establish relationships between tables
3. Relational database schema of a company is given in the following figure.

Relational Database Schema - COMPANY



EMPLOYEE
FNAME MINIT LNAME SSN BDATE ADDRESS SEX SALARY SUPERSSN DNO

DEPARTMENT
DNAME DNUMBER MGRSSN MGRSTARTDATE

DEPT_LOCATIONS
DNUMBER DLOCATION

PROJECT
PNAME PNUMBER PLOCATION DNUM

WORKS__ON
ESSN PNO HOURS

DEPENDENT
ESSN DEPENDENT_NAME SEX BDATE RELATIONSHIP

QUESTIONS TO BE PERFORMED ON ABOVE SCHEMA :

1. Create above tables with relevant **Primary Key, Foreign Key and other constraints**
2. Populate the tables with data
3. Display all the details of all employees working in the company.
4. Display **ssn, lname, fname, address** of employees who work in department no 7.
5. Retrieve the **Birthdate and Address** of the employee whose name is 'Franklin T. Wong'
6. Retrieve the name and salary of every employee
7. Retrieve all distinct salary values
8. Retrieve all employee names whose address is in 'Bellaire'
9. Retrieve all employees who were born during the 1950s
10. Retrieve all employees in department 5 whose salary is between 50,000 and 60,000(inclusive)
11. Retrieve the names of all employees who do not have supervisors
12. Retrieve SSN and department name for all employees
13. Retrieve the name and address of all employees who work for the 'Research' department
14. For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birth date.
15. For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.
16. Retrieve all combinations of Employee Name and Department Name
17. Make a list of all project numbers for projects that involve an employee whose last name is 'Narayan' either as a worker or as a manager of the department that controls the project.
18. Increase the salary of all employees working on the 'ProductX' project by 15%. Retrieve employee name and increased salary of these employees.
19. Retrieve a list of employees and the project name each works in, ordered by the employee's department, and within each department ordered alphabetically by employee first name.
20. Select the names of employees whose salary does not match with salary of any employee in department 10.
21. Retrieve the employee numbers of all employees who work on project located in Bellaire, Houston, or Stafford.
22. Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary. Display with proper headings.
23. Find the sum of the salaries and number of employees of all employees of the 'Marketing' department, as well as the maximum salary, the minimum salary, and the average salary in this department.
24. Select the names of employees whose salary is greater than the average salary of all employees in department 10.
25. Delete all dependents of employee whose **ssn is '123456789'**.
26. Perform a query using alter command to drop/add field and a constraint in Employee table.

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM
IV SEMESTER **COMPUTER SCIENCE** TIME: 4HRS/WEEK
CS 4603 (3) **OBJECT ORIENTED PROGRAMMING THROUGH JAVA**
MAX.MARKS:100
w.e.f. 20-21 admitted batch-"20AH" **SYLLABUS**

COURSE OBJECTIVES:

- Gain knowledge about basic Java language syntax and semantics to write Java programs and use concepts such as variables, conditional and iterative execution methods.
- Understand the fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc and exception handling mechanisms.
- Understand the principles of inheritance, packages and interfaces.

COURSE OUTCOMES :

- To recognize how to execute a simple as well as Java application and underlying the principles of Object-Oriented Programming.
- Describe and implement various Inheritance and Polymorphism forms using Java Classes and Interfaces.
- Implement efficient Java applets, exception handling and multithreading concepts in real life programming domains and hence enhance employability skills.

UNIT I : INTRODUCTION TO JAVA: Features of Java, The Java virtual Machine, Parts of Java.

NAMING CONVENTIONS AND DATA TYPES: Naming Conventions in Java, Data Types in Java, Literals

OPERATORS IN JAVA: Operators, Priority of Operators.

CONTROL STATEMENTS IN JAVA: if... else Statement, do... while Statement, while Loop, for Loop, switch Statement, break Statement, continue Statement, return Statement

INPUT AND OUTPUT: Accepting Input from the Keyboard, Reading Input with Java. util. Scanner Class, Displaying Output with System.out.printf(), Displaying Formatted Output with String. format()

ARRAYS: Types of Arrays, Three Dimensional Arrays (3D array), array name. length, Command Line Arguments

UNIT II : STRINGS: Creating Strings, String Class Methods, String Comparison, Immutability of Strings

INTRODUCTION TO OOPS: Problems in Procedure Oriented Approach, Features of Object-Oriented Programming System (OOPS)

CLASSES AND OBJECTS: Object Creation, Initializing the Instance Variables, Access Specifiers, Constructors

METHODS IN JAVA: Method Header or Method Prototype, Method Body, Understanding Methods, Static Methods, Static Block, The keyword 'this', Instance Methods, Passing Primitive Data Types to Methods, Passing Objects to Methods, Passing Arrays to Methods, Recursion, Factory Methods.

INHERITANCE: Inheritance, The keyword 'super', The Protected Specifier, Types of Inheritance

UNIT III : POLYMORPHISM: Polymorphism with Variables, Polymorphism using Methods, Polymorphism with Static Methods, Polymorphism with Private Methods, Polymorphism with Final Methods, final Class

TYPE CASTING: Types of Data Types, Casting Primitive Data Types, Casting Referenced Data Types, the Object Class

ABSTRACT CLASSES: Abstract Method and Abstract Class

INTERFACES: Interface, Multiple Inheritance using Interfaces

PACKAGES: Package, Different Types of Packages, The JAR Files, Interfaces in a Package, Creating Sub Package in a Package, Access Specifiers in Java, Creating API Document

EXCEPTION HANDLING: Errors in Java Program, Exceptions, throws Clause, throw Clause, Types of Exceptions, Re – throwing an Exception

UNIT IV : STREAMS: Stream, Creating a File using FileOutputStream, Reading Data from a File using FileInputStream, Creating a File using FileWriter, Reading a File using FileReader, Zipping and Unzipping Files, Serialization of Objects, Counting Number of Characters in a File, File Copy, File Class.

THREADS: Single Tasking, Multi Tasking, Uses of Threads, Creating a Thread and Running it, Terminating the Thread, Single Tasking Using a Thread, Multi Tasking Using Threads, Multiple Threads Acting on Single Object, Thread Class Methods, Deadlock of Threads, Thread Communication, Thread Priorities, thread Group, Daemon Threads, Applications of Threads, Thread Life Cycle.

UNIT V : APPLETS: Creating an Applet, Uses of Applets, <APPLET> tag, A Simple Applet, An Applet with Swing Components, Animation in Applets, A Simple Game with an Applet, Applet Parameters

JAVA DATABASE CONNECTIVITY: Database Servers, Database Clients, JDBC (Java Database Connectivity), Working with Oracle Database, Working with MySQL Database, Stages in a JDBC Program, Registering the Driver, Connecting to a Database, Preparing SQL Statements, Using jdbc–odbc Bridge Driver to Connect to Oracle Database, Retrieving Data from MySQL Database, Retrieving Data from MS Access Database, Stored Procedures and Callable Statements, Types of Result Sets

PRESCRIBED TEXT BOOK: Core Java: An Integrated Approach, Authored by Dr. R. Nageswara Rao & Kogent Learning Solutions Inc.

REFERENCE BOOKS:

- E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw- Hill Company.
- John R. Hubbard, Programming with Java, Second Edition, Schaum's outline Series, TMH.
- Deitel&amp;Deitel. Java TM: How to Program, PHI (2007).

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM
IV SEMESTER **COMPUTER SCIENCE** TIME: 2HRS/WEEK
CS 4653 (2) **OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB**
MAX.MARKS:50
w.e.f. 20-21 admitted batch-"20AH" **PRACTICALS SYLLABUS**

COURSE OBJECTIVES: To enable the students to:

- Understand the fundamentals of object oriented programming in java, including defining classes, invoking objects along with constructors, arrays and vectors.
- Discuss the principles of inheritance, interface and packages.

COURSE OUTCOMES:

After Completion of this course the student would be able to:

- Use an integrated development environment to write, compile, run and test simple object oriented java programs.
- Apply skills using basic control structures, arrays, object oriented principles including encapsulation and information hiding.
- Implement multithreaded programs and Exception handling.
- Apply the programming concepts as and when required in the future application development.

1. Write a program to read ***Student Name, Reg.No, Marks[5]*** and calculate ***Total***, ***Percentage, Result***. Display all the details of students

2. Write a program to perform the following String Operations
a. Read a string
b. Find out whether there is a given substring or not
c. Compare existing string by another string and display status
d. Replace existing string character with another character
e. Count number of works in a string

3. Java program to implements Addition and Multiplication of two N X N matrices.

4. Java program to demonstrate the use of Constructor.

5. Calculate area of the following shapes using method overloading.
a. Triangle
b. Rectangle
c. Circle
d. Square

6. Implement inheritance between ***Person (Aadhar, Surname, Name, DOB, and Age)*** and ***Student (Admission Number, College, Course, Year)***classes where ReadData(), DisplayData() are overriding methods.

7. Java program for implementing Interfaces

8. Java program on Multiple Inheritance.

9. Java program for to display ***Serial Number from 1 to N*** by creating two Threads

10. Java program to demonstrate the following exception handlings
a. Divided by Zero

b. Array Index Out of Bound
c. File Not Found
d. Arithmetic Exception
e. User Defined Exception

11. Create an Applet to display different shapes such as Circle, Oval, Rectangle, Square and Triangle.

12. Write a program to create ***Book (ISBN,Title, Author, Price, Pages, Publisher)***structure and store book details in a file and perform the following operations
a. Add book details
b. Search a book details for a given ISBN and display book details, if available
c. Update a book details using ISBN
d. Delete book details for a given ISBN and display list of remaining Books

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

IV SEMESTER **COMPUTER SCIENCE** Time: 4Hrs/Week

OPERATING SYSTEMS

20-21 admitted batch-"20AH" **SYLLABUS** Max.Marks:100

COURSE OBJECTIVES:

To enable the students to:

- Understand the overall structure and components of operating system.
- Analyze the key concept of Process Management and concurrency problem.
- Understand different approaches to memory management.

COURSE OUTCOMES:

Upon successful completion of the course, a student will be able to:

- Demonstrate the structure and design of operating systems.
- **Compare various algorithms for process scheduling.**
- **Apply various deadlock handling strategies to solve resource allocation problems.**
- **Evaluate the performance of different memory management techniques and page replacement algorithms and therefore develop employability skills.**
- Describe file concepts and analyse various disk scheduling strategies.

UNIT- I

What is Operating System? History and Evolution of OS, Basic OS functions, Resource Abstraction, Types of Operating Systems– Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real time Systems.

UNIT- II

Processor and User Modes, Kernels, System Calls and System Programs, System View of the Process and Resources, Process Abstraction, Process Hierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling, Non-Preemptive and Preemptive Scheduling Algorithms.

UNIT III

Process Management: Deadlock, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock, Deadlock Handling Approaches: Deadlock Prevention, Deadlock Avoidance and Deadlock Detection and Recovery.
Concurrent and Dependent Processes, Critical Section, Semaphores, Methods for Inter-process Communication; Process Synchronization, Classical Process Synchronization Problems: Producer-Consumer, Reader-Writer.

UNIT IV

Memory Management: Physical and Virtual Address Space; Memory Allocation Strategies– Fixed and -Variable Partitions, Paging, Segmentation, Virtual Memory.

UNIT V

File and I/O Management, OS security : Directory Structure, File Operations, File Allocation Methods, Device Management, Pipes, Buffer, Shared Memory, Security Policy Mechanism, Protection, Authentication and Internal Access Authorization
Introduction to Android Operating System, Android Development Framework, Android Application Architecture, Android Process Management and File System, Small Application Development using Android Development Framework.
https://nptel.ac.in/courses/106/105/106105214/
http://www.infocobuild.com/education/audio-video-courses/computer-science/OperatingSystems-IIT-Delhi/lecture-36.html
https://www.youtube.com/watch?v=AnGOeYJCv6s
https://www.youtube.com/watch?v=U1Jpvni0Aak
https://nptel.ac.in/content/storage2/courses/126104006/LectureNotes/Week-2_IntroductionToAndroid.pdf
https://www.youtube.com/watch?v=fzQcQV0UCUM

Prescribed Text Book:

Operating System Principles by Abraham Silberschatz, Peter Baer Galvin and Greg Gagne (7thEdition) Wiley India Edition.

Reference Books:

Operating Systems: Internals and Design Principles by Stallings (Pearson)
Operating Systems by J. Archer Harris (Author), Jyoti Singh (Author) (TMH)
Online Resources for UNIT V

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

IV SEMESTER **COMPUTER SCIENCE** Time: 2Hrs/Week

OPERATING SYSTEMS LAB USING C

20-21 admitted batch-"20AH" **SYLLABUS** Max.Marks:50

COURSE OBJECTIVES:

To enable the students to:

- Analyze the concept of Process Management and concurrency problem.
- Understand different approaches to memory management.

COURSE OUTCOMES:

Upon successful completion of the course, a student will be able to:

- Evaluate the performance of different types of CPU scheduling algorithms
- Compare different page replacement policies.
- Implement file organization techniques.
- Recognize need of Banker's algorithm for deadlock avoidance

1. Write a program to implement Round Robin CPU Scheduling algorithm
2. Simulate SJF CPU Scheduling algorithm
3. Write a program the FCFS CPU Scheduling algorithm
4. Write a program to Priority CPU Scheduling algorithm
5. Simulate Sequential file allocation strategies
6. Simulate Indexed file allocation strategies
7. Simulate Linked file allocation strategies
8. Simulate MVT and MFT memory management techniques
9. Simulate Single level directory File organization techniques
10. Simulate Two level File organization techniques
11. Simulate Hierarchical File organization techniques
12. Write a program for Bankers Algorithm for Dead Lock Avoidance
13. Implement Bankers Algorithm Dead Lock Prevention.
14. Simulate all Page replacement algorithms.
a) FIFO b) LRU c) LFU
15. Simulate Paging Technique of memory management

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

IV SEMESTER **COMPUTER SCIENCE** Time: 4Hrs/Week
CS 4604 (3) **OPERATING SYSTEMS** Max.Marks:100
w.e.f.20-21 admitted batch-"20AH" **SYLLABUS**

COURSE OBJECTIVES:

To enable the students to:

- Understand the overall structure and components of operating system.
- Analyze the key concept of Process Management and concurrency problem.
- Understand different approaches to memory management.

COURSE OUTCOMES:

Upon successful completion of the course, a student will be able to:

- Demonstrate the structure and design of operating systems.
- Compare various algorithms for process scheduling.
- Apply various deadlock handling strategies to solve resource allocation problems.
- Evaluate the performance of different memory management techniques and page replacement algorithms and therefore develop employability skills.
- Describe file concepts and analyse various disk scheduling strategies.

UNIT- I

What is Operating System? History and Evolution of OS, Basic OS functions, Resource Abstraction, Types of Operating Systems– Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real time Systems.

UNIT- II

Processor and User Modes, Kernels, System Calls and System Programs, System View of the Process and Resources, Process Abstraction, Process Hierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling, Non-Preemptive and Preemptive Scheduling Algorithms.

UNIT III

PROCESS MANAGEMENT: Deadlock, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock, Deadlock Handling Approaches: Deadlock Prevention, Deadlock Avoidance and Deadlock Detection and Recovery.

Concurrent and Dependent Processes, Critical Section, Semaphores, Methods for Inter-process Communication; Process Synchronization, Classical Process Synchronization Problems: Producer-Consumer, Reader-Writer.

UNIT IV

MEMORY MANAGEMENT: Physical and Virtual Address Space; Memory Allocation Strategies– Fixed and -Variable Partitions, Paging, Segmentation, Virtual Memory.

UNIT V

FILE AND I/O MANAGEMENT, OS SECURITY: Directory Structure, File Operations, File Allocation Methods, Device Management, Pipes, Buffer, Shared Memory, Security Policy Mechanism, Protection, Authentication and Internal Access Authorization

Introduction to Android Operating System, Android Development Framework, Android Application Architecture, Android Process Management and File System, Small Application Development using Android Development Framework.

https://nptel.ac.in/courses/106/105/106105214/
http://www.infocobuild.com/education/audio-video-courses/computer-science/OperatingSystems-IIT-Delhi/lecture-36.html
https://www.youtube.com/watch?v=AnGOeYJCv6s
https://www.youtube.com/watch?v=U1Jpvni0Aak
https://nptel.ac.in/content/storage2/courses/126104006/LectureNotes/Week-2_IntroductionToAndroid.pdf
https://www.youtube.com/watch?v=fzQcQV0UCUM

PRESCRIBED TEXT BOOK:

Operating System Principles by Abraham Silberschatz, Peter Baer Galvin and Greg Gagne (7thEdition) Wiley India Edition.

REFERENCE BOOKS:

Operating Systems: Internals and Design Principles by Stallings (Pearson)
Operating Systems by J. Archer Harris (Author), Jyoti Singh (Author) (TMH)
Online Resources for UNIT V

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM
IV SEMESTER **COMPUTER SCIENCE** Time: 2Hrs/Week
CS 4654 (2) **OPERATING SYSTEMS LAB USING "C"** Max.Marks:50
w.e.f.20-21 admitted batch-"20AH" **SYLLABUS**

COURSE OBJECTIVES:

To enable the students to:

- Analyze the concept of Process Management and concurrency problem.
- Understand different approaches to memory management.

COURSE OUTCOMES:

Upon successful completion of the course, a student will be able to:

- Evaluate the performance of different types of CPU scheduling algorithms
- Compare different page replacement policies.
- Implement file organization techniques.
- Recognize need of Banker's algorithm for deadlock avoidance

1. Write a program to implement Round Robin CPU Scheduling algorithm

2. Simulate SJF CPU Scheduling algorithm

3. Write a program the FCFS CPU Scheduling algorithm

4. Write a program to Priority CPU Scheduling algorithm

5. Simulate Sequential file allocation strategies

6. Simulate Indexed file allocation strategies

7. Simulate Linked file allocation strategies

8. Simulate MVT and MFT memory management techniques

9. Simulate Single level directory File organization techniques

10. Simulate Two level File organization techniques

11. Simulate Hierarchical File organization techniques

12. Write a program for Bankers Algorithm for Dead Lock Avoidance

13. Implement Bankers Algorithm Dead Lock Prevention.

14. Simulate all Page replacement algorithms.
    a) FIFO b) LRU c) LFU
15. Simulate Paging Technique of memory management

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

V SEMESTER **COMPUTER SCIENCE** Time: 3Hrs/Week

CS-E3-5603(3) **DATA SCIENCE**

Max.Marks:100

w.e.f. 20AHBatch **SYLLABUS**

COURSE OBJECTIVES: -

- To learn the basics of data science
- Implement it in real-world environment
- Learn basic programming using data science.

COURSE OUTCOMES:

- Students after successful completion of the course will be able to:
- Develop programming abilities.
- Demonstrate statistical analysis of data.
- Develop the ability to build data-based models.
- Improve their skill in data management.
- Apply data science concepts and methods to solve problems in real-world contexts.

UNIT - I: Introduction: The Ascendance of Data, What is Data Science? , Finding key Connectors, Data Scientists You May Know, Salaries and Experience, Paid Accounts, Topics of Interest, Onward.

Python: Getting Python, The Zen of Python, Whitespace Formatting, Modules, Arithmetic, Functions, Strings, Exceptions, Lists, Tuples, Dictionaries, Sets, Control Flow, Truthiness, Sorting, List Comprehensions, Generators and Iterators, Randomness, Object – Orienting Programming, Functional Tools, enumerate, zip and Argument Unpacking, args and kwargs, Welcome to Data Sciencester!

Visualizing Data: matplotlib, Bar charts, Line charts, Scatterplots.

Linear Algebra: Vectors, Matrices

UNIT -II: Statistics: Describing a Single Set of Data, Correlation, Simpson's Paradox, some Other Correlation Caveats, Correlation and Causation.

Probability: Dependence and Independence, Conditional Probability, Bayes's Theorem, Random Variables, Continuous Distributions, The Normal Distribution, The Central Limit Theorem.

Hypothesis and Inference: Statistical Hypothesis Testing, Example: Flipping a Coin, Confidence Intervals, P-hacking, Example: Running an A/B Test, Bayesian Inference.

Gradient Descent: The Idea behind Gradient Descent, Estimating the Gradient, Using the Gradient, Choosing the Right Step Size, Putting It All Together, Stochastic Gradient Descent.

UNIT -III :Getting Data: stdin and stdout, Reading Files – The Basics of Text Files, Delimited Files, Scraping the Web - HTML and the parsing Thereof, Example:

O'Reilly Books About Data, Using APIs – JSON (and XML), Using an Unauthenticated API, Finding APIs.

Working with Data: Exploring Your Data, Exploring One-Dimensional Data, Two Dimensions Many Dimensions, Cleaning and Munging, Manipulating Data, Rescaling, Dimensionality Reduction.

Machine Learning: Modeling, What Is Machine Learning? Over fitting and under fitting, Correctness, The Bias-Variance Trade-off, Feature Extraction and selection.

UNIT-IV: K-Nearest Neighbors: The Model, Example: Favorite Languages, The Curse of Dimensionality.

Naive Bayes: A Really Dumb Spam Filter, A More Sophisticated Spam Filter, Implementation, Testing Our Model.

Simple Linear Regression: The Model, Using Gradient Descent, Maximum Likelihood Estimation.

Multiple Regression: The Model, Further Assumptions of the Least Squares Model, Fitting the Model, Interpreting the Model, Goodness of Fit.

UNIT–V: Logistic Regression: The Problem, The Logistic Function, Applying the Model, Goodness of Fit Support Vector Machines.

Decision Trees: What Is a Decision Tree? Entropy, The Entropy of a Partition, Creating a Decision Tree, Putting It All Together, Random Forests.

Neural Networks: Perceptron, Feed-Forward Neural Networks And Back propagation, Example: Defeating a CAPTCHA.

Clustering: The Idea, The Model, Example: Meetups , Choosing k, Example: Clustering Colors, Bottom-up Hierarchical Clustering.

REFERENCES:

1. Data Science from Scratch by Joel Grus O'Reilly Media .
2. Wes McKinney, "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython", O'Reilly, 2nd Edition, 2018.
3. Jake VanderPlas, "Python Data Science Handbook: Essential Tools for Working with Data", O'Reilly, 2017.
4. Web resources:
   a. https://www.edx.org/course/analyzing-data-with-python
   b.http://math.ecnu.edu.cn/~lfzhou/seminar/[Joel_Grus]_Data_Science_from_Scratch_First_Princ.pdf
5. Other web sources suggested by the teacher concerned and the college librarian including reading material.

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

V SEMESTER **COMPUTER SCIENCE** Time: 3Hrs/Week

CS-E3-5604(3) **PYTHON FOR DATA SCIENCE** Max.Marks:100

w.e.f. 20AH Batch **SYLLABUS**

COURSE OBJECTIVES: -

- To learn various python concepts, functions and packages
- To enable data manipulation and analysis.
- To learn how to solve real-world problems using python.

COURSE OUTCOMES:

Students after successful completion of the course will be able to:

- Learning the need for data science using python.
- Solve basic problems using Python built-in data types and their methods.
- To demonstrate an application with user-defined modules and packages using OOP concept
- Implement data operations using NumPy arrays for easy storage.
- Applying data manipulation methods using Pandas.
- Implement data pre-processing and visualization of applications using Pandas

UNIT-I: Introduction to Data Science - Why Python? - Essential Python libraries - Python Introduction- Features, Identifiers, Reserved words, Indentation, Comments, Built-in Data types and their Methods: Strings, List, Tuples, Dictionary, Set - Type Conversion- Operators.

Decision Making- Looping- Loop Control statement- Math and Random number functions. User defined functions - function arguments & its types.

UNIT–II: User defined Modules and Packages in Python- Files: File manipulations, File and Directory related methods - Python Exception Handling.

OOPs Concepts -Class and Objects, Constructors – Data hiding- Data Abstraction- Inheritance.

UNIT–III :NumPy Basics: Arrays and Vectorized Computation- The NumPyndarray- Creating ndarrays- Data Types for ndarrays- Arithmetic with NumPy Arrays- **Basic Indexing and Slicing** - Boolean Indexing-Transposing Arrays and Swapping Axes.

Universal Functions: Fast Element-Wise Array Functions- Mathematical and Statistical Methods-Sorting- Unique and Other Set Logic.

UNIT–IV: Introduction to pandas Data Structures: Series, Data Frame and Essential Functionality: Dropping Entries- Indexing, Selection, and Filtering- Function Application and Mapping- Sorting and Ranking.

Summarizing and Computing Descriptive Statistics- Unique Values, Value Counts, and Membership. Reading and Writing Data in Text Format.

UNIT–V: Data Cleaning and Preparation: Handling Missing Data - Data Transformation: Removing Duplicates, Transforming Data Using a Function or Mapping, Replacing Values, Detecting and Filtering Outliers- String Manipulation: Vectorized String Functions in pandas. Plotting with pandas: Line Plots, Bar Plots, Histograms and Density Plots, Scatter or Point Plots.

References: -

1. Y. Daniel Liang, "Introduction to Programming using Python", Pearson, 2012.
2. Wes McKinney, "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython", O'Reilly, 2nd Edition, 2018.
3. Jake VanderPlas, "Python Data Science Handbook: Essential Tools for Working with Data", O'Reilly, 2017.
4. Wesley J. Chun, "Core Python Programming", Prentice Hall, 2006.
5. Mark Lutz, "Learning Python", O'Reilly, 4th Edition, 2009.
6. Web resources:

a. https://www.edx.org/course/python-basics-for-data-science

b. https://www.edx.org/course/analyzing-data-with-python

c. https://www.coursera.org/learn/python-plotting?specialization=data-science-python

d. https://www.programmer-books.com/introducing-data-science-pdf/

e. https://www.cs.uky.edu/~keen/115/Haltermanpythonbook.pdf

7. Other web sources suggested by the teacher concerned and the college librarian including reading material.

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM
V SEMESTER **COMPUTER SCIENCE** Time: 3Hrs/Week
CS-E3-5653(2) **DATA SCIENCE LAB** Max.Marks:50
w.e.f. 20AH Batch **PRACTICAL SYLLABUS**

COURSE OBJECTIVES:

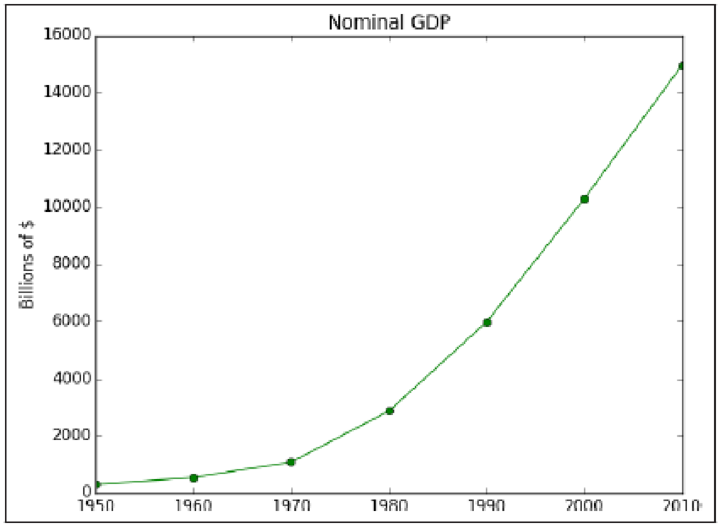
- Understand the basics of Data Science
- To learn how to solve real-world problems using python
- Gain knowledge about the installation of python software

COURSE OUTCOMES: On successful completion of this practical course, student shall be able to:

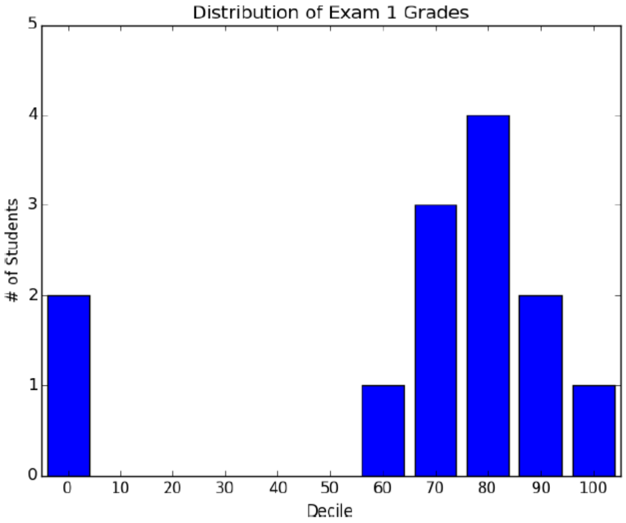
- Apply data science solutions to real world problems.
- Implement the programs to get the required data and present the outputs using Python language.
- Execute statistical analysis with open python software.

PRACTICAL (LABORATORY) SYLLABUS:

1. Write a Python program to create a line chart for values of year and GDP as given below



2.Write a Python program to create a bar chart to display number of students secured different grading as given below



2. Write a Python program to create a time series chart by taking one year month wise stock data in a CSV file.
3. Write a Python program to plot distribution curve.
4. Import a CSV file and perform various Statistical and Comparison operations on rows/columns. Write a python program to plot a graph of people with pulse rate p vs. height h. The values of P and H are to be entered by the user.
5. Import rainfall data of some location with the help of packages available in R Studio and plot a chart of your choice.

** ** **

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

V SEMESTER **COMPUTER SCIENCE** Time: 2Hrs/Week

CS-E3-5654(2) **PYTHON FOR DATA SCIENCE LAB** Max.Marks:50

w.e.f. 20AH Batch **SYLLABUS**

COURSE OBJECTIVES:

- Understand the basics and syntax of Python programming
- To learn how to solve real-world problems using python
- Gain knowledge about the installation of python software

COURSE OUTCOMES:

On successful completion of this practical course, student shall be able to:

- Implement simple programs in Python.
- Demonstrate programs related to various structures like arrays, lists, Data frames, etc.
- Implement programs related and using files.
- Implement applications related to data science.

Practical (Laboratory) Syllabus:

1. Perform Creation, indexing, slicing, concatenation and repetition operations on Python built-in data types: Strings, List, Tuples, Dictionary, Set.
2. Apply Python built-in data types: Strings, List, Tuples, Dictionary, Set and their methods to solve any given problem.
3. Handle numerical operations using math and random number functions.
4. Create user-defined functions with different types of function arguments.
5. Create packages and import modules from packages.
6. Perform File manipulations- open, close, read, write, append and copy from one file to another.
7. Write a program for Handle Exceptions using Python Built-in Exceptions
8. Write a program to implement OOP concepts like Data hiding and Data Abstraction.

9. Create NumPy arrays from Python Data Structures, Intrinsic NumPy objects and Random Functions.

10. Manipulation of NumPy arrays- Indexing, Slicing, Reshaping, Joining and Splitting.

11. Computation on NumPy arrays using Universal Functions and Mathematical methods.

12. Load an image file and do crop and flip operation using NumPy Indexing.

13. Create Pandas Series and Data Frame from various inputs.

14. Import any CSV file to Pandas Data Frame and perform the following:

    (a) Visualize the first and last 10 records

    (b) Get the shape, index and column details

    (c) Select/Delete the records (rows)/columns based on conditions.

    (d) Perform ranking and sorting operations.

    (e) Do required statistical operations on the given columns.

    (f) Find the count and uniqueness of the given categorical values.

    (g) Rename single/multiple columns

15. Import any CSV file to Pandas Data Frame and perform the following:

    (a) Handle missing data by detecting and dropping/ filling missing values.

    (b) Transform data using apply () and map() method.

    (c) Detect and filter outliers.

    (d) Perform Vectorized String operations on Pandas Series.

    (e) Visualize data using Line Plots, Bar Plots, Histograms, Density Plots and Scatter Plots.

** ** **