Vivekanand Education Society's

College of Arts, Science and Commerce

(Autonomous)

Sindhi Society, Chembur, Mumbai, Maharashtra – 400 071.

Accredited by NAAC "A Grade" in 3rd Cycle - 2017
Best College Award – Urban Area, University of Mumbai (2012-13)
Recipient of FIST Grant (DST) and STAR College Grant (DBT)

Affiliated to the

University of Mumbai

Syllabus for

Program: B.Sc. (Information Technology)

(Program code: VESUSIT)

As per Choice Based Semester and Grading System (CBSGS)
with effect from Academic Year 2023 - 2024

S.Y.B.Sc. Information Technology

(SEMESTER III)

Course Code	Title	Credits & Lectures per Semester	Lectures per Week
VESUIT201	Python Programming	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUIT202	Data Structures	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUIT203	Computer Networks	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUIT204	Database Management Systems	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	

VESUIT205	Applied Mathematics	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUITP201	Python Programming Practical	01	02
VESUITP202	Data Structures Practical	01	02
VESUITP203	Computer Networks Practical	01	02
VESUITP 204	Database Management Systems Practical	01	02
VESUITP205	Mobile Programming Practical	01	02

(SEMESTER IV)

Course Code	Title	Credits & Lectures per Semester	Lectures per Week
VESUIT206	Core Java	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUIT207	Embedded Systems	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUIT208	Computer Oriented Statistical Methods	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUIT209	Software Engineering	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	

	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUIT210	Computer Graphics and Animation	03	03
	Unit I :	9 Lectures	
	Unit II:	9 Lectures	
	Unit III :	9 Lectures	
	Unit IV:	9 Lectures	
	Unit V :	9 Lectures	
VESUITP206	Core Java Practical	01	02
VESUITP207	Advanced Web Programming Practical	01	02
VESUITP208	Computer Oriented Statistical Methods Practical	01	02
VESUITP209	Software Engineering Practical	01	02
VESUITP210	Computer Graphics and Animation Practical	01	02

Course title: Python Programming
Course code: VESUIT201

Objective: To understand and develop programs using python Language

Learning Outcomes (LO):

LO1: Develop the basics of programming skills using Python Language
LO2: Understand and make effective use of python data structures.
LO3: Develop OOP programming skills using Python
LO4: Design and develop GUI based applications in Python

Unit	Details	Lectures
I	**Introduction:** The Python Programming Language, History, features, Installing Python, Running Python program, Debugging: Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages, The Difference Between Brackets, Braces, and Parentheses, **Variables and Expressions:** Values and Types, Variables, Variable Names and Keywords, Type conversion, Operators and Operands, Expressions, Interactive Mode and Script Mode, Order of Operations. **Conditional Statements:** if, if-else, nested if –else **Looping**: for, while, nested loops **Control statements:** Terminating loops, skipping specific conditions	9 Lectures
II	**Functions:** Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution, Parameters and Arguments, Variables and Parameters Are Local, Stack Diagrams, Fruitful Functions and Void Functions, Why Functions?Importing with from, Return Values, Incremental Development, Composition, Boolean Functions, More Recursion, Leap of Faith, Checking Types **Strings:** A String Is a Sequence, Traversal with a for Loop, String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, format() method of string, The in Operator, String Comparison, String Operations.	9 Lectures

III	**Lists:** Values and Accessing Elements, Lists are mutable, traversing a List, Deleting elements from List, Built-in List Operators, Concatenation, Repetition, In Operator, Built-in List functions and methods **Tuples and Dictionaries:** Tuples, Accessing values in Tuples, Tuple Assignment, Tuples as return values, Variable-length argument tuples, Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in Tuple Functions Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods **Files:** Text Files, The File Object Attributes, Directories **Exceptions:** Built-in Exceptions, Handling Exceptions, Exception	9 Lectures
	with Arguments, User-defined Exceptions	
IV	**Regular Expressions –** Concept of regular expression, various types of regular expressions, using match function. **Classes and Objects:** Overview of OOP (Object Oriented Programming), Class Definition, Creating Objects, Instances as Arguments, Instances as return values, Built-in Class Attributes, Inheritance, Method Overriding, Data Encapsulation, Data Hiding **Multithreaded Programming:** Thread Module, creating a thread, synchronizing threads, multithreaded priority queue **Modules:** Importing module, Creating and exploring modules, Math module, Random module, Time module, Date module, OS module	9 Lectures
V	**Creating the GUI Form and Adding Widgets:** **Widgets:** Button, Canvas,Checkbutton, Entry, Frame, Label, Listbox, Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel, Spinbox, PanedWindow, LabelFrame, tkMessagebox. Handling Standard attributes and Properties of Widgets. **Layout Management:** Designing GUI applications with proper Layout Management features. **Look and Feel Customization**:Enhancing Look and Feel of GUI using different appearances of widgets. **Storing Data in Our MySQL Database via Our GUI :**Connecting to a MySQL database from Python, Configuring the MySQL connection, Designing the Python GUI database, Using the INSERT command, Using the UPDATE command, Using the DELETE command, Storing and retrieving data from MySQL database. **Pandas Library:** Introduction, series, dataframe, reading from CSV files, analyzing dataframes.	9 Lectures

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Think Python	Allen Downey	O'Reilly	1St	2012
2.	An Introduction to Computer Science using Python 3	JasonMontojo, Jennifer Campbell, Paul Gries	SPD	1St	2014
3.	Python GUI Programming Cookbook	Burkhard A. Meier	Packt		2015
4.	Introduction to Problem Solving with Python	E. Balagurusamy	TMH	1St	2016
5.	Murach's Python programming	Joel Murach, Michael Urban	SPD	1St	2017
6.	Object-oriented Programming in Python	Michael H. Goldwasser, David Letscher	Pearson Prentice Hall	1St	2008
7.	Exploring Python	Budd	TMH	1St	2016

List of Practical	
1.	**Write the program for the following:**
a.	Create a program that asks the user to enter their name and their age. Print out a message addressed to them that tells them the year that they will turn 100 years old.
b.	Enter the number from the user and depending on whether the number is even or odd, print out an appropriate message to the user.
c.	Write a program to generate the Fibonacci series.
d.	Write a function that reverses the user defined value.
e.	Write a function to check the input value is Armstrong and also write the function for Palindrome.
f.	Write a recursive function to print the factorial for a given number.
2.	**Write the program for the following:**
a.	Write a function that takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise.
b.	Define a function that computes the *length* of a given list or string.
c.	Define a *procedure*histogram() that takes a list of integers and prints a histogram to the screen. For example, histogram([4, 9, 7]) should print the following: **** ********* *******
3.	**Write the program for the following:**

a.	A *pangram* is a sentence that contains all the letters of the English alphabet at least once, for example: *The quick brown fox jumps over the lazy dog*. Your task here is to write a function to check a sentence to see if it is a pangram or not.
b.	Take a list, say for example this one: a=[1,1,2,3,5,8,13,21,34,55,89] and write a program that prints out all the elements of the list that are less than 5.
4.	**Write the program for the following:**
a.	Write a program that takes two lists and returns True if they have at least one common member.
b.	Write a Python program to print a specified list after removing the 0th, 2nd, 4th and 5th elements.
c.	Write a Python program to clone or copy a list
5.	**Write the program for the following:**
a.	Write a Python script to sort (ascending and descending) a dictionary by value.
b.	Write a Python script to concatenate following dictionaries to create a new one. Sample Dictionary : dic1={1:10, 2:20} dic2={3:30, 4:40} dic3={5:50,6:60} Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
c.	Write a Python program to sum all the items in a dictionary.
6.	**Write the program for the following:**
a.	Write a Python program to read an entire text file.
b.	Write a Python program to append text to a file and display the text.
c.	Write a Python program to read last n lines of a file.
7.	**Write the program for the following:**
a.	Design a class that store the information of student and display the same
b.	Implement the concept of inheritance using python
c.	Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). i. Write a method called add which returns the sum of the attributes x and y. ii. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. iii. Write a static method called subtract, which takes two number parameters, b and c, and returns b - c. iv. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and a deleter

	for manipulating the values of x and y.
8.	**Write the program for the following:**
a.	Open a new file in IDLE (“New Window” in the “File” menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the “Control Flow and Functions” exercise into this file and save it. Now open a new file and save it in the same directory. You should now be able to importyour own module like this: importgeometry Try and add print dir(geometry) to the file and run it. Now write a function pointyShapeVolume(x, y, squareBase) that calculates the volume of a square pyramid if squareBase is True and of a right circular cone if squareBase is False. x is the length of an edge on a square if squareBase is True and the radius of a circle when squareBase is False. y is the height of the object. First use squareBase to distinguish the cases. Use the circleArea and squareArea from the geometry module to calculate the base areas.
b.	Write a program to implement exception handling.
9.	**Write the program for the following:**
a.	Try to configure the widget with various options like: bg=”red”, family=”times”, size=18
b.	Try to change the widget type and configuration options to experiment with other widget types like Message, Button, Entry, Checkbutton, Radiobutton, Scale etc.
10.	**Design the database applications for the following:**
a.	Design a simple database application that stores the records and retrieve the same.
b.	Design a database application to search the specified record from the database.
c.	Design a database application to that allows the user to add, delete and modify the records.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Think Python	Allen Downey	O’Reilly	1st	2012
2.	An Introduction to Computer Science using Python 3	JasonMontojo, Jennifer Campbell, Paul Gries	SPD	1st	2014

Course title: Data Structures
Course code: VESUIT202

Objective: To understand and develop applications using different data structures

Learning Outcomes (LO):

LO1: Implement abstract data types using arrays and linked list.
LO2: Apply the different linear data structures like stack and queue to various computing problems.
LO3: Implement different types of trees and apply them to problem solutions.
LO4: Discuss graph structure and understand various operations on graphs and their applicability.
LO5: Analyze the various sorting and searching algorithms.
LO6: Understand the hashing technique and hash functions.

Unit	Details	Lectures
I	**Introduction:** Data and Information, Data Structure, Classification of Data Structures, Primitive Data Types, Abstract Data Types, Data structure vs. File Organization, Operations on Data Structure, Algorithm, Importance of Algorithm Analysis, Complexity of an Algorithm, Asymptotic Analysis and Notations, Big O Notation, Big Omega Notation, Big Theta Notation, Rate of Growth and Big O Notation. **Array**: Introduction, One Dimensional Array, Memory Representation of One Dimensional Array, Traversing, Insertion, Deletion, Searching, Sorting, Merging of Arrays, Multidimensional Arrays, Memory Representation of Two Dimensional Arrays, General Multi-Dimensional Arrays, Sparse Arrays, SparseMatrix, Memory Representation of Special kind of Matrices, Advantages and Limitations of Arrays.	9 Lectures
II	Linked List: Linked List, One-way Linked List, Traversal of Linked List, Searching, Memory Allocation and De-allocation, Insertion in Linked List, Deletion from Linked List, Copying a List into Other List, Merging Two Linked Lists, Splitting a List into Two Lists, Reversing One way linked List, Circular Linked List, Applications of Circular Linked List, Two way Linked List, Traversing a Two way Linked List, Searching in a Two way linked List, Insertion of an element in Two way Linked List, Deleting a node from Two way Linked List, Header Linked List, Applications of the Linked list, Representation of Polynomials, Storage of Sparse Arrays,Implementing other Data Structures.	9 Lectures

III	**Stack**: Introduction, Operations on the Stack Memory Representation of Stack, Array Representation of Stack, Applications of Stack, Evaluation of Arithmetic Expression, Matching Parenthesis, infix and postfix operations, Recursion**.** **Queue:** Introduction, Queue, Operations on the Queue, Memory Representation of Queue, Array representation of queue, Linked List Representation of Queue, Circular Queue,Some special kinds of queues, Deque,Priority Queue, Application of Priority Queue, Applications of Queues.	9 Lectures
IV	**Sorting and Searching Techniques** Bubble, Selection, Insertion, Merge Sort. Searching: Sequential,	9 Lectures
	Binary, Indexed Sequential Searches, Binary Search. **Tree:**Tree,Binary Tree, Properties of Binary Tree, Memory Representation of Binary Tree, Operations Performed on Binary Tree,Reconstruction of Binary Tree from its Traversals, Huffman Algorithm, Binary Search Tree, Operations on Binary Search Tree, Heap, Memory Representation of Heap, Operation on Heap, Heap Sort. **Advanced Tree Structures:**Red Black Tree, Operations Performed on Red Black Tree, AVL Tree, Operations performed on AVL Tree, 2-3 Tree, B-Tree.	
V	**Hashing Techniques** Hash function, Address calculation techniques, Common hashing functions Collision resolution, Linear probing, Quadratic,Double hashing, Buckethashing, Deletion and rehashing **Graph**: Introduction, Graph, Graph Terminology, Memory Representation of Graph, Adjacency Matrix Representation of Graph, Adjacency List or Linked Representation of Graph, Operations Performed on Graph, GraphTraversal, Applications of the Graph, Reachability, Shortest Path Problems, Spanning Trees.	9 Lectures

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A Simplified Approach to Data Structures	Lalit Goyal, Vishal Goyal, Pawan Kumar	SPD	1st	2014
2.	An Introduction to Data Structure with Applications	Jean – Paul Tremblay and Paul Sorenson	Tata MacGraw Hill	2nd	2007
3.	Data Structure and Algorithm	Maria Rukadikar	SPD	1st	2017

4.	Schaum's Outlines Data structure	Seymour Lipschutz	Tata McGraw Hill	2nd	2005
5.	Data structure – A Pseudocode Approach with C	AM Tanenbaum, Y Langsamand MJ Augustein	Prentice Hall India	2nd	2006
6.	Data structure andAlgorithm Analysis in C	Weiss, Mark Allen	Addison Wesley	1st	2006

List of Practical	
1.	**Implement the following:**
a.	Write a program to store the elements in 1-D array and perform the operations like searching, sorting and reversing the elements. [Menu Driven]
b.	Read the two arrays from the user and merge them and display the elements in sorted order.[Menu Driven]
c.	Write a program to perform the Matrix addition, Multiplication and Transpose Operation. [Menu Driven]
2.	**Implement the following for Linked List:**
a.	Write a program to create a single linked list and display the node elements in reverse order.
b.	Write a program to search the elements in the linked list and display the same
c.	Write a program to create double linked list and sort the elements in the linked list.
3.	**Implement the following for Stack:**
a.	Write a program to implement the concept of Stack with Push, Pop, Display and Exit operations.
b.	Write a program to convert an infix expression to postfix and prefix conversion.
c.	Write a program to implement Tower of Hanoi problem.
4.	**Implement the following for Queue:**
a.	Write a program to implement the concept of Queue with Insert, Delete, Display and Exit operations.
b.	Write a program to implement the concept of Circular Queue
c.	Write a program to implement the concept of Deque.
5.	**Implement the following sorting techniques:**
a.	Write a program to implement bubble sort.
b.	Write a program to implement selection sort.
c.	Write a program to implement insertion sort.

6.	**Implement the following data structure techniques:**
a.	Write a program to implement merge sort.
b.	Write a program to search the element using sequential search.
c.	Write a program to search the element using binary search.
7.	**Implement the following data structure techniques:**
a.	Write a program to create the tree and display the elements.
b.	Write a program to construct the binary tree.
c.	Write a program for inorder, postorder and preorder traversal of tree
8.	**Implement the following data structure techniques:**
a.	Write a program to insert the element into maximum heap.
b.	Write a program to insert the element into minimum heap.
9.	**Implement the following data structure techniques:**
a.	Write a program to implement the collision technique.
b.	Write a program to implement the concept of linear probing.
10.	**Implement the following data structure techniques:**
a.	Write a program to generate the adjacency matrix.
b.	Write a program for shortest path diagram.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Data Structures and Algorithms Using Python	RanceNecaise	Wiley	First	2016
2.	Data Structures Using C and C++	Langsam,Augenstein, Tanenbaum	Pearson	First	2015

Course title: Computer Networks
Course code: VESUIT203

Objective: To understand different networking concept and configure the networks

Learning Outcomes (LO):

LO1: Understanding the principles of networking through protocol layering and using bottom-up approach
LO2: Understanding the data communication process
LO3: Understanding the TCP/IP protocol suite
LO4: Awareness of wired and wireless network

Unit	Details	Lectures
I	**Introduction:** Data communications, networks, network types, Internet history, standards and administration. **Network Models:** Protocol layering, TCP/IP protocol suite, The OSI model. **Introduction to Physical layer:** Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance. **Digital and Analog transmission:** Digital-to-digital conversion, analog-to-digital conversion, transmission modes, digital-to-analog conversion, analog-to-analog conversion.	9 Lectures
II	**Bandwidth Utilization: Multiplexing and Spectrum Spreading:** Multiplexing, Spread Spectrum **Transmission media:** Guided Media**,** Unguided Media **Switching:** Introduction, circuit switched networks, packet switching, structure of a switch. **Introduction to the Data Link Layer:** Link layer addressing, Data Link Layer Design Issues, Error detection and correction, block coding, cyclic codes, checksum, forward error correction, error correcting codes, error detecting codes.	9 Lectures
III	**Data Link Control:** DLC services, data link layer protocols, HDLC, Point-to-point protocol. **Media Access Control**: Random access, controlled access, channelization, Wired LANs – Ethernet Protocol, standard ethernet, fast ethernet, gigabit ethernet, 10 gigabit ethernet, **Wireless LANs:** Introduction, IEEE 802.11 project, Bluetooth, WiMAX, Cellular telephony, Satellite networks. **Connecting devices and Virtual LANs.**	9 Lectures

IV	**Introduction to the Network Layer**: Network layer services, packet switching, network layer performance, IPv4 addressing, forwarding of IP packets, Internet Protocol, ICMPv4, Mobile IP **Unicast Routing:**Introduction, routing algorithms, unicast routing protocols. **Next generation IP:** IPv6 addressing, IPv6 protocol, ICMPv6 protocol, transition from IPv4 to IPv6.	9 Lectures
V	**Introduction to the Transport Layer**: Introduction, Transport layer protocols (Simple protocol, Stop-and-wait protocol, Go-Back-n protocol, Selective repeat protocol, Bidirectional protocols), Transport	9 Lectures
	layer services, User datagram protocol, Transmission control protocol **Standard Client-Server Protocols:**World wide-web and HTTP, FTP, Electronic mail, Telnet, Secured Shell, Domain name system.	

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Data Communication and Networking	Behrouz A. Forouzan	Tata McGraw Hill	Fifth Edition	2013
2.	TCP/IP Protocol Suite	Behrouz A. Forouzan	Tata McGraw Hill	Fourth Editio n	2010
3.	Computer Networks	Andrew Tanenbaum	Pearson	Fifth	2013

List of Practical

1. IPv4 Addressing and Subnetting

a) Given an IP address and network mask, determine other information about the IP addresssuch as:
- Network address
- Network broadcast address
- Total number of host bits
- Number of hosts

b) Given an IP address and network mask, determine other information about the IP addresssuch as:
- The subnet address of this subnet
- The broadcast address of this subnet
- The range of host addresses for this subnet
- The maximum number of subnets for this subnet mask
- The number of hosts for each subnet
- The number of subnet bits
- The number of this subnet

2.	Use of ping and tracert / traceroute, ipconfig / ifconfig, route and arp utilities.
3.	Configure IP static routing.
4.	Configure IP routing using RIP.
5.	Configuring Simple OSPF.
6.	Configuring DHCP server and client.
7.	Create virtual PC based network using virtualization software and virtual NIC.
8.	Configuring DNS Server and client.
9.	Configuring OSPF with multiple areas.
10.	Use of Wireshark to scan and check the packet information of following protocols ● HTTP ● ICMP ● TCP ● SMTP ● POP3

Course Title: Database Management System

Course Code: VESUIT204

Objective:

The main objective of this course is to enable students to understand the fundamental concepts of database analysis and design.

Learning Outcomes (LO):

On successful completion of this course students will be able to

LO 1: Define program-data independence, data models for database systems, database schema and database instances.
LO 2: Recall Relational Algebra concepts, and use it to translate queries to Relational Algebra statements and vice versa.
LO 3: Identify Structure Query Language statements used in creation and manipulation of Database.
LO 4: Understand the concept of transaction, concurrency and recovery.
LO 5: Solve Database problems using PL/SQL which will include the use of Procedures, Functions, Packages, and Triggers.

Unit	Details	Lectures
I	**Introduction to Databases and Transactions** What is database system, purpose of database system, view of data, relational databases, database architecture, **Data Models** The importance of data models, Basic building blocks, Business rules, The evolution of data models, Degrees of data abstraction. **Database Design,ER Diagram and Unified Modeling Language** Database design and ER Model:overview, ERModel, Constraints, ERDiagrams, ERDIssues, weak entity sets, Codd's rules, Relational Schemas, Introduction to UML	9 Lectures
II	**Relational database model:** Logical view of data, keys, integrity rules, Relational Database design: features of good relational database design, atomic domain and Normalization (1NF, 2NF, 3NF, BCNF). **Relational Algebra and Calculus** Relational algebra: introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison. **Calculus:** Tuple relational calculus, Domain relational Calculus, calculus algebra, computational capabilities	9 Lectures
III	**Constraints, Views and SQL** Constraints, types of constraints, Integrity constraints, Views: Introduction to views, data independence, security, updates on views, comparison between tables and views SQL: data definition, aggregate function, Null Values, nested sub queries, Joined relations. Triggers.	9 Lectures

IV	**Transaction management and Concurrency** Control Transaction management: ACID properties (Transaction Management) serializability and concurrency control, Time stamping methods, optimistic methods, database recovery management. User Managed Backups Recovery Manager Backups - RMAN , Transaction Recovery, System Recovery	9 Lectures
V	**PL-SQL**: Beginning with PL / SQL,Identifiers and Keywords, Operators, Expressions, Sequences,Control Structures, Cursors and Transaction,Collections and composite data types, Procedures and Functions, Exceptions Handling,Packages,With Clause and Hierarchical Retrieval,Triggers.	9 Lectures

References :

Database System and Concepts by A Silberschatz, H Korth, S Sudarshan, Tata McGraw Hill, 6th Edition.
Database Systems by Peter Rob and Carlos Coronel, Cengage Learning, 9th Edition
Programming with PL/SQL for Beginners by H.Dand, R.Patil and T. Sambare, X –Team, First Edition

Additional References:

Introduction to Database System by C.J.Date, Pearson, First Edition

List of Practical	
1.	**SQL Statements – 1**
a.	Writing Basic SQL SELECT Statements
b.	Restricting and Sorting Data
c.	Single-Row Functions
2.	**SQL Statements – 2**
a.	Displaying Data from Multiple Tables
b.	Aggregating Data Using Group Functions
c.	Subqueries
3.	**Manipulating Data**
a.	Using INSERT statement
b.	Using DELETE statement
c.	Using UPDATE statement
4.	**Creating and Managing Tables**
a.	Creating and Managing Tables
b.	Including Constraints

5.	**Creating and Managing other database objects**
a.	Creating Views
b.	Other Database Objects
c.	Controlling User Access
6.	**Using SET operators, Date/Time Functions, GROUP BY clause (advanced features) and advanced subqueries**
a.	Using SET Operators
b.	Datetime Functions
c.	Enhancements to the GROUP BY Clause
d.	Advanced Subqueries
7.	**PL/SQL Basics**
a.	Declaring Variables
b.	Writing Executable Statements
c.	Interacting with the Oracle Server
d.	Writing Control Structures
8.	**Composite data types, cursors and exceptions.**
a.	Working with Composite Data Types
b.	Writing Explicit Cursors
c.	Handling Exceptions
9.	**Procedures and Functions**
a.	Creating Procedures
b.	Creating Functions
c.	Managing Subprograms
d.	Creating Packages
10.	**Creating Database Triggers**

Course Title: Applied Mathematics
Course Code: VESUIT205

Objective: To develop fundamental mathematical skills and the ability for independent mathematical learning and reasoning which will familiarize them with the concepts required for the competitive exams

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1: Able to apply mathematical concepts and principles like matrices, linear equations to perform computations
LO2: Solve problems based on complex numbers and linear differential equations, multiple integrals and apply the concepts of integration
LO3: Use the Method of Laplace transforms to solve initial-value problems for linear differential equations with constant coefficients.

Unit	Details	Lectures
I	**Matrices:** Definition of matrices, Properties of matrices, Elementary Transformation, Rank of Matrix, Echelon or Normal Matrix, Inverse of matrix, Systems of Linear equations, Linear dependence and linear independence of vectors, Linear transformation, Characteristics roots and characteristics vectors, Properties of characteristic vectors, Cayley Hamilton Theorem, Similarity of matrices, Reduction of matrix to a diagonal matrix which has elements as characteristics values.	9 Lectures
II	**Complex Numbers:** Complex number, Equality of complex numbers, Graphical representation of complex number(Argand's Diagram), Polar form of complex numbers, Polar form of x+iy for different signs of x,y, Exponential form of complex numbers, Mathematical operation with complex numbers and their representation on Argand's Diagram, Circular functions of complex angles, Definition of hyperbolic function, Relations between circular and hyperbolic functions, Inverse hyperbolic functions, Differentiation and Integration, Graphs of the hyperbolic functions, Logarithms of complex quality, j(=i)as an operator(Electrical circuits) .	9 Lectures
III	**Equation of the first order and of the first degree:** Separation of variables, Equations homogeneous in x and y, Non-homogeneous linear equations, exact differential Equation, Integrating Factor, Linear Equation and equation reducible to this form, Method of substitution. **Differential equation of the first order of a degree higher than the first:**	9 Lectures

	Introduction, Solvable for p (or the method of factors), Solve for y, Solve for x, Clairaut's form of the equation, Methods of Substitution, Method of Substitution. **Linear Differential Equations with Constant Coefficients:**Introduction, The Differential Operator, Linear Differential Equation f(D) y = 0, Different cases depending on the nature of the root of the equation f(D) = 0, Linear differential equation f(D) y = X, The complementary Function, The inverse operator 1/f(D) and the symbolic expiration for the particular integral 1/f(D) X; the general methods, Particular integral : Short methods, Particular integral : Other methods, Differential equations reducible to the linear differential equations with constant coefficients.	
IV	**The Laplace Transform:** Introduction, Definition of the Laplace Transform, Table of Elementary Laplace Transforms, Theorems on Important Properties of Laplace Transformation, First Shifting Theorem, Second Shifting Theorem, The Convolution Theorem, Laplace Transform of an Integral, Laplace Transform of Derivatives, **Inverse Laplace Transform:** Shifting Theorem, Partial fraction Methods, Use of Convolution Theorem, Solution of Ordinary Linear Differential Equations with Constant Coefficients, Solution of Simultaneous Ordinary Differential Equations, Laplace Transformation of Special Function, Periodic Functions, Heaviside Unit Step Function, Dirac-delta Function(Unit Impulse Function),	9 Lectures
V	**Multiple Integrals:** Double Integral, Change of the order of the integration, Double integral in polar coordinates, Triple integrals. **Applications of integration:** Areas, Volumes of solids. **Beta and Gamma Functions –** Definitions**,**Properties and Problems. Duplication formula.	9 Lectures

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A text book of Applied Mathematics Vol I	P. N. Wartikar and J. N. Wartikar	Pune VidyathiGraha		
2.	Applied Mathematics II	P. N. Wartikar and J. N. Wartikar	Pune VidyathiGraha		
3.	Higher Engineering Mathematics	Dr. B. S. Grewal	Khanna Publications		

Mobile Programming

The practical's will be based on HTML5, CSS, CORDOVA and PhoneGAP API. (Android will be introduced later after they learn Java)

List of Practical	
	Setting up CORDOVA, PhoneGAP Project and environment.
1.	Creating and building simple "Hello World" App using Cordova Adding and Using Buttons Adding and Using Event Listeners
2.	● Creating and Using Functions ● Using Events ● Handlingand Using Back Button
3.	● Installing and Using Plugins ● Installing and Using Battery Plugin ● Installing and Using Camera Plugin
4.	● Installing and Using Contacts Plugin ● Installing and Using Device Plugin ● Installing and Using Accelerometer Plugin
5.	● Install and Using Device Orientation plugin ● Install and Using Device Orientation plugin ● Create and Using Prompt Function
6.	Installing and Using File Plugin Installing and Using File Transfer Plugin Using Download and Upload functions
7.	● Installing and Using Globalization Plugin ● Installing and Using Media Plugin ● Installing and Using Media Capture Plugin
8.	● Installing and Using Network Information Plugin
	● Installing and Using Splash Screen Plugin ● Installing and Using Vibration Plugin
9.	● Developing Single Page Apps ● Developing Multi-page Apps ● Storing Data Locally in a Cordova App

10.	● Use of sqlite plugin with PhoneGap / apache Cordova ● Using Sqlite read/write and search ● Populating Cordova SQLite storage with the JQuery API

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Apache Cordova 4 Programming	John M. Wargo	Addison-Wesley Professional	1st	2015
2.	Apache Cordova in Action	Raymond Camden	Manning Publications	1st	2015
3.	PhoneGap By Example	Andrey Kovalenko	PACKT Publishing	1st	2015

SEMESTER 4

Course title: Core Java
Course code: VESUIT206

Objective: To understand and develop programs using C Language

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1: Use Object Oriented Programming concepts for problem solving using Java
LO2: Understand and make effective use of special features of java.
LO3: Design and develop GUI based applications.

Unit	Details	Lectures
I	**Introduction:** History, architecture and its components, Java Class File, Java Runtime Environment, The Java Virtual Machine, JVM Components, The Java API, java platform, java development kit, Lambda Expressions, Methods References, Type Annotations, Method Parameter Reflection, setting the path environment variable, Java Compiler And Interpreter, java programs, java applications, main(), public, static, void, string[] args, statements, white space, case sensitivity, identifiers, keywords, comments, braces and code blocks, variables, variable name **Data types:** primitive data types, Object Reference Types, Strings, Auto boxing, operators and properties of operators, Arithmetic operators, assignment operators, increment and decrement operator, relational operator, logical operator, bitwise operator, conditional operator.	9 Lectures
II	**Control Flow Statements:** The If…Else If…Else Statement, The Switch…Case Statement **Iterations:** The While Loop, The Do … While Loop, The For Loop, The Foreach Loop, Labeled Statements, The Break And Continue Statements, The Return Statement **Classes:** Types of Classes, Scope Rules, Access Modifier, Instantiating Objects From A Class, Initializing The Class Object And Its Attributes, Class Methods, Accessing A Method, Method Returning A Value, Method's Arguments, Method Overloading, Variable Arguments [Varargs], Constructors, this Instance, super Instance, Characteristics Of Members Of A Class, constants, this instance, static fields of a class, static methods of a class, garbage collection.	9 Lectures

III	**Inheritance:** Derived Class Objects, Inheritance and Access Control, Default Base Class Constructors, this and super keywords. Abstract Classes And Interfaces, Abstract Classes, Abstract Methods, Interfaces, What Is An Interface? How Is An Interface Different From An Abstract Class?, Multiple Inheritance, Default Implementation, Adding New Functionality, Method Implementation, Classes V/s Interfaces, Defining An Interface, Implementing Interfaces. **Packages:** Creating Packages, Default Package, Importing Packages, Using A Package.	9 Lectures
IV	Enumerations, Arrays: Two Dimensional Arrays, Multi-Dimensional Arrays, Vectors, Adding Elements To A Vector, Accessing Vector Elements, Searching For Elements In A Vector, Working With The Size of The Vector. Multithreading: the thread control methods, thread life cycle, the main thread, creating a thread, extending the thread class. Exceptions: Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause Byte streams: reading console input, writing console output, reading file, writing file, writing binary data, reading binary data, getting started with character streams, writing file, reading file	9 Lectures
V	Event Handling: Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes and inner classes. Abstract Window Toolkit: Window Fundamentals, Component, Container, Panel, Window, Frame, Canvas. Components – Labels, Buttons, Check Boxes, Radio Buttons, Choice Menus, Text Fields, Text, Scrolling List, Scrollbars, Panels, Frames Layouts: Flow Layout, Grid Layout, Border Layout, Card Layout. Swing: JFrames, Lists , Tables, Trees, Text Components, Progress Indicators, Component Organizers	9 Lectures

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Core Java 8 for Beginners	Vaishali Shah, Sharnam Shah	SPD	1st	2015
2.	Java: The Complete Reference	Herbert Schildt	McGraw Hill	9th	2014
3.	Murach's beginning Java with Net Beans	Joel Murach , Michael Urban	SPD	1st	2016
4.	Core Java, Volume I: Fundamentals	Hortsman	Pearson	9th	2013

5.	Core Java, Volume II: Advanced Features	Gary Cornell and Hortsman	Pearson	8th	2008
6.	Core Java: An Integrated Approach	R. Nageswara Rao	DreamTech	1st	2008

List of Practical	
1.	**Java Basics**
a.	Write a Java program that takes a number as input and prints its multiplication table upto 10.
b.	Write a Java program to display the following pattern. ***** **** *** ** *
c.	Write a Java program to print the area and perimeter of a circle.
2.	**Use of Operators**
a.	Write a Java program to add two binary numbers.
b.	Write a Java program to convert a decimal number to binary number and vice versa.
c.	Write a Java program to reverse a string.
3.	**Java Data Types**
a.	Write a Java program to count the letters, spaces, numbers and other characters of an input string.
b.	Implement a Java function that calculates the sum of digits for a given char array consisting of the digits '0' to '9'. The function should return the digit sum as a long value.
c.	Find the smallest and largest element from the array
4.	**Methods and Constructors**
a.	Designed a class SortData that contains the method asec() and desc().
b.	Designed a class that demonstrates the use of constructor and destructor.
c.	Write a java program to demonstrate the implementation of abstract class.
5.	**Inheritance**
a.	Write a java program to implement single level inheritance.
b.	Write a java program to implement method overriding
c.	Write a java program to implement multiple inheritance.

6.	Packages and Arrays
a.	Create a package, Add the necessary classes and import the package in java class.
b.	Write a java program to add two matrices and print the resultant matrix.
c.	Write a java program for multiplying two matrices and print the product for the same.
7.	Vectors and Multithreading
a.	Write a java program to implement the vectors.
b.	Write a java program to implement thread life cycle.
c.	Write a java program to implement multithreading.
8.	File Handling
a.	Write a java program to open a file and display the contents in the console window.
b.	Write a java program to copy the contents from one file to other file.
c.	Write a java program to read the student data from user and store it in the file.
9.	GUI and Exception Handling
a.	Design a AWT program to print the factorial for an input value.
b.	Design an swing program to perform various string operations like reverse string, string concatenation etc.
c.	Write a java program to implement exception handling.
10.	GUI Programming.
a.	Design an swing application that contains the interface to add student information and display the same.

b.	Design a calculator based on swing application.
c.	Design an swing application to generate result marks sheet.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Core Java 8 for Beginners	Vaishali Shah, Sharnam Shah	SPD	1st	2015
2.	Java: The Complete Reference	Herbert Schildt	McGraw Hill	9th	2014
3.	Murach's beginning Java with Net Beans	Joel Murach , Michael Urban	SPD	1st	2016
4.	Core Java, Volume I: Fundamentals	Hortsman	Pearson	9th	2013
5.	Core Java, Volume II: Advanced Features	Gary Cornell and Hortsman	Pearson	8th	2008
6.	Core Java: An Integrated Approach	R. Nageswara Rao	DreamTech	1st	2008

Course title: Embedded Systems
Course code: VESUIT207

Objective: To understand and develop programs using C Language

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1: Define embedded systems and identify applications to real word systems
LO2: Utilize hardware, software, and peripherals involved in an embedded system
LO3: Understand basic microprocessor and microcontroller functionality utilizing registers and memory and Hardware/Software interfacing concepts
LO4: Write programs Related to Embedded functionality

Unit	Details	Lectures
I	Introduction: Embedded Systems and general purpose computer systems, history, classifications, applications and purpose of embedded systems Core of embedded systems: microprocessors and microcontrollers, RISC and CISC controllers, Big endian and Little endian processors, Application specific ICs, Programmable logic devices, COTS, sensors and actuators, communication interface, embedded firmware, other system components. Characteristics and quality attributes of embedded systems: Characteristics, operational and non-operational quality attributes	9 Lectures
II	Embedded Systems – Application and Domain Specific: Application specific – washing machine, domain specific - automotive. Embedded Hardware: Memory map, i/o map, interrupt map, processor family, external peripherals, memory – RAM , ROM, types of RAM and ROM, memory testing, CRC ,Flash memory. Peripherals: Control and Status Registers, Device Driver, Timer Driver - Watchdog Timers.	9 Lectures

III	The 8051 Microcontrollers:Microcontrollers and Embedded processors, Overview of 8051 family.8051 Microcontroller hardware, Input/output pins, Ports, and Circuits, External Memory. 8051 Programming in C: Data Types and time delay in 8051 C, I/O Programming, Logic operations, Data conversion Programs.	9 Lectures
IV	Designing Embedded System with 8051 Microcontroller:Factors to be considered in selecting a controller, why 8051 Microcontroller, Designing with 8051. Programming embedded systems: structure of embedded program, infinite loop, compiling, linking and debugging.	9 Lectures
V	Real Time Operating System (RTOS):Operating system basics, types of operating systems, Real-Time Characteristics, Selection Process of an RTOS. Design and Development: Embedded system development Environment – IDE, types of file generated on cross compilation, disassembler/ de-compiler, simulator, emulator and debugging, embedded product development life-cycle, trends in embedded industry.	9 Lectures

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Programming Embedded Systems in C and C++	Michael Barr	O'Reilly	First	1999
2.	Introduction to embedded systems	Shibu K V	Tata Mcgraw-Hill	First	2012
3.	The 8051Microcontroller and Embedded Systems	Muhammad Ali Mazidi	Pear son	Second	2011

4.	Embedded Systems	Rajkamal	Tata Mcgraw -Hill		

List of Practical	
1.	Design and develop a reprogrammable embedded computer using 8051 microcontrollers and to show the following aspects. a. Programming b. Execution c. Debugging
2. A	Configure timer control registers of 8051 and develop a program to generate given time delay.
B	To demonstrate use of general purpose port i.e. Input/ output port of two controllers for data transfer between them.
3. A	Port I / O: Use one of the four ports of 8051 for O/P interfaced to eight LED's.Simulate binary counter (8 bit) on LED's
B	To interface 8 LEDs at Input-output port and create different patterns.
C	To demonstrate timer working in timer mode and blink LED without using any loop delay routine.
4. A	Serial I / O: Configure 8051 serial port for asynchronous serial communication with serial port of PC exchange text messages to PC and display on PC screen. Signify end of message by carriage return.
B	To demonstrate interfacing of seven-segment LED display and generate counting from 0 to 99 with fixed time delay.
C	Interface 8051 with D/A converter and generate square wave of given frequency on oscilloscope.
5. A	Interface 8051 with D/A converter and generate triangular wave of given frequency on oscilloscope.
B	Using D/A converter generate sine wave on oscilloscope with the help of lookup table stored in data area of 8051.

6.	Interface stepper motor with 8051 and write a program to move the motor through a given angle in clock wise or counter clock wise direction.
7.	Generate traffic signal.
8.	Implement Temperature controller.
9.	Implement Elevator control.
10.	Using FlashMagic
A	To demonstrate the procedure for flash programming for reprogrammable embedded system board using FlashMagic
B	To demonstrate the procedure and connections for multiple controllers programming of same type of controller with same source code in one go, using flash magic.

Course Title : Computer Oriented Statistical Methods
Course Code : VESUIT208

Objective:
To demonstrate understanding of numerical and statistical methods in support of the analysis, design and application for problem solving in the field of information technology.

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1: Able to apply mean, median, mode, standard deviation on any given data and work with R Language.
LO2: Compare Skewness, Kurtosis, probability ,sampling theory and apply statistical estimation theory and statistical decision theory
LO3: Identify the role of chi-square test for real data and apply curve fitting, method of least squares and correlation theory for any given data.

Unit	Details	Lectures
I	**The Mean, Median, Mode, and Other Measures of Central Tendency**: Index, or Subscript, Notation, Summation Notation, Averages, or Measures of Central Tendency ,The Arithmetic Mean , The Weighted Arithmetic Mean ,Properties of the Arithmetic Mean, The Arithmetic Mean Computed from Grouped Data ,The Median ,The Mode, The Empirical Relation Between the Mean, Median, and Mode, The Geometric Mean G, The Harmonic Mean H ,The Relation Between the Arithmetic, Geometric, and Harmonic Means, The Root Mean Square, Quartiles, Deciles, and Percentiles, Software and Measures of Central Tendency. **Measures of Dispersion**: Dispersion, or Variation, The Range, The Mean Deviation, The Semi- Interquartile Range, The 10–90 Percentile Range, The Standard Deviation, The Variance, Short Methods for Computing the Standard Deviation, Properties of the Standard Deviation, Charlie's Check, Sheppard's Correction for Variance, Empirical Relations Between Measures of Dispersion, Absolute and Relative Dispersion; Coefficient of Variation, Standardized Variable; Standard Scores, Software and Measures of Dispersion. **Introduction to R:** Basic syntax, data types, variables, operators, control statements, R-functions, R –Vectors, R – lists, R Arrays.	9 Lectures

II	**Moments, Skewness, and Kurtosis** : Moments , Moments for Grouped Data ,Relations Between Moments , Computation of Moments for Grouped Data, Charlie's Check and Sheppard's Corrections, Moments in Dimensionless Form, Skewness, Kurtosis, Population Moments, Skewness, and Kurtosis, Software Computation of Skewness and Kurtosis. **Elementary Probability Theory**: Definitions of Probability, Conditional Probability; Independent and Dependent Events, Mutually Exclusive Events, Probability Distributions, Mathematical Expectation, Relation Between Population, Sample Mean, and Variance, Combinatorial Analysis, Combinations, Stirling's Approximation to n!, Relation of Probability to Point Set Theory, Euler or Venn Diagrams and Probability. **Elementary Sampling Theory :** Sampling Theory, Random Samples and Random Numbers, Sampling With and Without Replacement, Sampling Distributions, Sampling Distribution of Means, Sampling Distribution of Proportions, Sampling Distributions of Differences and Sums, Standard Errors, Software Demonstration of Elementary Sampling Theory	9 Lectures
III	**Statistical Estimation Theory:** Estimation of Parameters, Unbiased Estimates, Efficient Estimates, Point Estimates and Interval Estimates; Their Reliability, Confidence-Interval Estimates of Population Parameters, Probable Error. **Statistical Decision Theory:** Statistical Decisions, Statistical Hypotheses, Tests of Hypotheses and Significance, or Decision Rules, Type I and Type II Errors, Level of Significance, Tests Involving Normal Distributions, Two-Tailed and One-Tailed Tests, Special Tests, OperatingCharacteristic Curves; the Power of a Test, p-Values for Hypothesis Tests, Control Charts, Tests Involving Sample Differences, Tests Involving Binomial Distributions. **Statistics in R**: mean, median, mode, Normal Distribution , Binomial Distribution, Frequency Distribution in R	9 Lectures
IV	**Small Sampling Theory**: Small Samples, Student's t Distribution, Confidence Intervals, Tests of Hypotheses and Significance, The Chi- Square Distribution, Confidence Intervals for Sigma , Degrees of Freedom, The F Distribution. **The Chi-Square Test:** Observed and Theoretical Frequencies, Definition of chi-square, Significance Tests, The Chi-Square Test for Goodness of Fit, Contingency Tables, Yates' Correction for Continuity, Simple Formulas for Computing chi-square, Coefficient of Contingency, Correlation of Attributes, Additive Property of chi-square.	9 Lectures

V	**Curve Fitting and the Method of Least Squares:** Relationship Between Variables, Curve Fitting, Equations of Approximating Curves, Freehand Method of Curve Fitting, The Straight Line, The Method of Least Squares, The Least-Squares Line, Nonlinear Relationships, The Least-Squares Parabola, Regression, Applications to Time Series, Problems Involving More Than Two Variables. **Correlation Theory:** Correlation and Regression, Linear Correlation, Measures of Correlation, The Least-Squares Regression Lines, Standard Error of Estimate, Explained and Unexplained Variation, Coefficient of Correlation, Remarks Concerning the Correlation Coefficient, Product-Moment Formula for the Linear Correlation Coefficient, Short Computational Formulas, Regression Lines and the Linear Correlation Coefficient, Correlation of Time Series, Correlation of Attributes, Sampling Theory of Correlation, Sampling Theory of Regression.	9 Lectures

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	STATISTICS	Murray R. Spiegel, Larry J. Stephens.	McGRAW – HILL INTERNATIONAL	FOURTH	
2.	A Practical Approach using R	R.B. Patil, H.J. Dand and R. Bhavsar	SPD	1st	2017
3.	FUNDAMENTAL OF MATHEMATICAL STATISTICS	S.C. GUPTA and V.K. KAPOOR	SULTAN CHAND and SONS	ELEVENTH REVISED	2011
4.	MATHEMATICAL STATISTICS	J.N. KAPUR and H.C. SAXENA	S. CHAND	TWENTIETH REVISED	2005

List of Practical

1.	Using R execute the basic commands, array, list and frames.
2.	Create a Matrix using R and Perform the operations addition, inverse, transpose and multiplication operations.
3.	Using R Execute the statistical functions: mean, median, mode, quartiles, range, inter quartile range histogram
4.	Using R import the data from Excel / .CSV file and Perform the above functions.
5.	Using R import the data from Excel / .CSV file and Calculate the standard deviation, variance,

	co-variance.
6.	Using R import the data from Excel / .CSV file and draw the skewness.
7.	Import the data from Excel / .CSV and perform the hypothetical testing.
8.	Import the data from Excel / .CSV and perform the Chi-squared Test.
9.	Using R perform the binomial and normal distribution on the data.
10.	Perform the Linear Regression using R.
11.	Compute the Least squares means using R.
12.	Compute the Linear Least Square Regression

Course Title: Software Engineering

Course Code: VESUIT209

Objective:

The main objective is to help students to develop skills that will enable them to construct high quality software that is reliable, and that is reasonably easy to understand, modify and maintain.

Learning Outcomes (LO):

On successful completion of this course students will be able to

LO1: To apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment.
LO2: Define and manage the overall scope of the project, documenting project goals, deliverables, constraints, performance criteria and resource requirements in consultation with project stakeholders and to understand quality control and how to ensure good quality software.
LO3: Understand the approaches of verification and validation including static analysis, and reviews
LO4: An ability to apply design and development principles in the construction of software systems of varying complexity.

Unit	Details	Lectures
I	**Introduction:** What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc. **Software Requirements:** Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements. **Software Processes:** Process and Project, Component Software Processes. **Software Development Process Models.** ● Waterfall Model. ● Prototyping. ● Iterative Development. ● Rational Unified Process. ● The RAD Model	9 Lectures

	• Time boxing Model • Spiral Model • Devops Model **Agile software development**: Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.	
II	**Socio-technical system:** Essential characteristics of socio technical systems, Emergent System Properties, Systems Engineering, Components of systems such as organization, people and computers, Dealing Legacy Systems. **Critical system:** Types of critical system, A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems. **Requirements Engineering Processes:** Feasibility study, Requirements elicitation and analysis, Requirements Validations, Requirements Management. **System Models:** Models and its types, Context Models, Behavioural Models, Data Models, Object Models, Structured Methods.	9 Lectures
III	**Architectural Design:** Architectural Design Decisions, System Organization, Modular Decomposition Styles, Control Styles, Reference Architectures. **User Interface Design:** Need of UI design, Design issues, The UI design Process, User analysis, User Interface Prototyping, Interface Evaluation. **Project Management:** Software Project Management, Management activities, Project Planning, Project Scheduling, Risk Management **Quality Management:** Process and Product Quality, Quality assurance and Standards, Quality Planning, Quality Control, Software Measurement and Metrics	9 Lectures
IV	**Verification and Validation:** Planning Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods. **Software Testing:** System Testing, Component Testing, Test Case Design, Test Automation. **Software Measurement:** Size-Oriented Metrics, Function-Oriented Metrics, Extended Function Point Metrics **Software Cost Estimation:** Software Productivity, Estimation Techniques, Algorithmic Cost Modelling, Project Duration and Staffing.	9 Lectures

V	**Process Improvement: Process and product quality,** Process Classification, Process Measurement, Process Analysis and Modeling, Process Change, The CMMI Process Improvement Framework. **Service Oriented Software Engineering:** Services as reusable components, Service Engineering, Software Development with Services. **Software reuse:** The reuse landscape, Application frameworks, Software product lines, COTS product reuse. **Distributed software engineering:** Distributed systems issues, Client– server computing, Architectural patterns for distributed systems, Software as a service	9 Lectures

References:

Software Engineering by Ian Somerville, Pearson Education, 9th Edition
Software Engineering by Pankaj Jalote, Narosa Publication
Software engineering, a practitioner's approach, By Roger Pressman, Tata McGraw Hill, 7th Edition.
UML A Beginners Guide by Jason T. Roff, McGraw Hill Professional 2003

Other References:

Software Engineering principles and practice by WS Jawadekar, Tata McGraw Hill.
Software Engineering- A Concise Study by S.A Kelkar, PHI India.
Software Engineering Concept and Applications by Subhajit Datta, Oxford Higher Education
Software Design by D.Budgen, Pearson education, 2nd Edition
Software Engineering by KL James, PHI
Object - Oriented Modeling and Design by Michael Blaha, James Rumbaugh, Pearson 2001
The unified modeling language user guide by Grady Booch, James Rumbaugh, Ivar Jacobson, Addison- Wesley 2006.
Learning UML 2. 0 by Kim Hamilton, Russ Miles, O'Reilly Media, 2006

List of Practical (To be executed using StarUML or any similar software)	
1.	Study and implementation of class diagrams.
2.	Study and implementation of Use Case Diagrams.
3.	Study and implementation of Entity Relationship Diagrams.
4.	Study and implementation of Sequence Diagrams.
5.	Study and implementation of State Transition Diagrams.

6.	Study and implementation of Data Flow Diagrams.
7.	Study and implementation of Collaboration Diagrams.
8.	Study and implementation of Activity Diagrams.
9.	Study and implementation of Component Diagrams.
10.	Study and implementation of Deployment Diagrams.

Course title: Computer Graphics
Course code: VESUIT210

Objective: To understand the basics of computer graphics and its software standards. The course provides theoretical background and develops programming skills in computer graphics.

Learning Outcomes (LO):
On successful completion of this course students will be able to:

LO1: Understand the basics of computer graphics, different graphics systems and applications of computer graphics.
LO2: Discuss various algorithms for scan conversion and filling of basic objects. Extract scene with different clipping methods and its transformation to graphics display device.
LO3: Use of geometric transformations on graphics objects and their application in composite form.
LO4: Explore projections and visible surface detection techniques. Render projected objects to naturalize the scene in 2D view and use of illumination models for this.

Unit	Details	Lectures
I	**Introduction to Computer Graphics:** Overview of Computer Graphics, Computer Graphics Application and Software, Description of some graphics devices, Input Devices for Operator Interaction, Active and Passive Graphics Devices, Display Technologies, Storage Tube Graphics Displays, Calligraphic Refresh Graphics Displays, Raster Refresh (Raster-Scan) Graphics Displays, Cathode Ray Tube Basics, Color CRT Raster Scan Basics, Video Basics, The Video Controller, Random-Scan Display Processor, LCD displays. **Scan conversion** – Digital Differential Analyzer (DDA) algorithm, Bresenhams' Line drawing algorithm. Bresenhams' method of Circle drawing, Midpoint Circle Algorithm, Midpoint Ellipse Algorithm, Mid-point criteria, Problems of Aliasing, end-point ordering and clipping lines, Scan Converting Circles, Clipping Lines algorithms– Cyrus-Beck, Cohen-Sutherland and Liang-Barsky, Clipping Polygons, problem with multiple components.	9 Lectures

II	**Two-Dimensional Transformations:** Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations. **Three-Dimensional Transformations:** Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-Dimensional Translation, Multiple Transformation, Rotation about an Arbitrary Axis in Space, Reflection through an Arbitrary Plane, Matrix Representation of 3D Transformations, Composition of 3D Transformations, Affine and Perspective Geometry, Perspective Transformations, Techniques for Generating Perspective Views, Vanishing Points, the Perspective Geometry and camera models, Orthographic Projections, Axonometric Projections, Oblique Projections, View volumes for projections.	9 Lectures
III	**Light:** Radiometry, Transport, Equation, Photometry **Color:** Colorimetry, Color Spaces, Chromatic Adaptation, Color Appearance **Visible-Surface Determination:** Techniques for efficient Visible-Surface Algorithms, Categories of algorithms, Back face removal, The z-Buffer Algorithm, Scan-line method, Painter's algorithms (depth sorting), Area sub-division method, BSP trees, Visible-Surface Ray Tracing, comparison of the methods	9 Lectures
IV	**Illumination models and surface rendering:** **Light sources, Basic Illumination Model, Displaying Light intensities, Halftone Patterns and dithering techniques, Polygon rendering methods, Ray tracing** **Plane Curves and Surfaces:** Curve Representation, Nonparametric Curves, Parametric Curves, Parametric Representation of a Circle, Parametric Representation of an Ellipse, Parametric Representation of a Parabola, Parametric Representation of a Hyperbola, Representation of Space Curves, Cubic Splines, , Bezier Curves, B-spline Curves, B-spline Curve Fit, B-spline Curve Subdivision, Parametric Cubic Curves, Quadric Surfaces. Bezier Surfaces.	9 Lectures

V	**Computer Animation:** Principles of Animation, Key framing, Deformations, Character Animation, Physics-Based Animation, Procedural Techniques, Groups of Objects. **Image Manipulation and Storage:** What is an Image? Digital image file formats, Image compression standard – JPEG, Image Processing - Digital image enhancement, contrast stretching, Histogram Equalization, smoothing and median Filtering.	9 Lectures

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Computer Graphics - Principles and Practice	J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes	Pearson	2nd	
2.	Steve Marschner, Peter Shirley	Fundamentals of Computer Graphics	CRC press	4th	2016
3.	Computer Graphics	Hearn, Baker	Pearson	2nd	
4.	Principles of Interactive Computer Graphics	William M. Newman and Robert F. Sproull	TMH	2nd	
5.	Mathematical Elements for CG	D. F. Rogers, J. A. Adams	TMH	2nd	

List of Practical	
1.	**Solve the following:**
a.	Study and enlist the basic functions used for graphics in C / C++ / Python language. Give an example for each of them.
b.	Draw a co-ordinate axis at the center of the screen.
2.	**Solve the following:**
a.	Divide your screen into four region, draw circle, rectangle, ellipse and half ellipse in each region with appropriate message.
b.	Draw a simple hut on the screen.
3.	**Draw the following basic shapes in the center of the screen :**
	i. Circle ii. Rectangle iii. Square iv. Concentric Circles v. Ellipse vi. Line
4.	**Solve the following:**

a.	Develop the program for DDA Line drawing algorithm.
b.	Develop the program for Bresenham's Line drawing algorithm.
5.	**Solve the following:**
a.	Develop the program for the mid-point circle drawing algorithm.
b.	Develop the program for the mid-point ellipse drawing algorithm.
6.	**Solve the following:**
a.	Write a program to implement 2D scaling.
b.	Write a program to perform 2D translation
7.	**Solve the following:**
a.	Perform 2D Rotation on a given object.
b.	Program to create a house like figure and perform the following operations. i. Scaling about the origin followed by translation. ii. Scaling with reference to an arbitrary point. iii. Reflect about the line y = mx + c.
8.	Solve the following:
a.	Write a program to implement Cohen-Sutherland clipping.
b.	Write a program to implement Liang - Barsky Line Clipping Algorithm
9.	Solve the following:
a.	Write a program to fill a circle using Flood Fill Algorithm.
b.	Write a program to fill a circle using Boundary Fill Algorithm.
10.	Solve the following:
a.	Develop a simple text screen saver using graphics functions.
b.	Perform smiling face animation using graphic functions.
c.	Draw the moving car on the screen.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Computer Graphics - Principles and Practice	J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes	Pearson Education	Second Edition	
2.	Steve Marschner, Peter Shirley	Fundamentals of Computer Graphics	CRC press	Fourth Edition	2016
3.	Computer Graphics	Hearn, Baker	Pearson Education	Second	
4.	Principles of Interactive Computer Graphics	William M. Newman and Robert F. Sproull	Tata McGraw Hill	Second	

Modality of assessment

The performance of the learners shall be evaluated into two parts. The learner's performance shall be assessed by Internal Assessment with 25% marks in the first part & by conducting the Semester End Examinations with 75% marks in the second part. Practical Examination will consist of End Sem examination.

Student will have to score 40% of marks in Internal assessment as well as End Sem examination to pass the course.

The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below: -

Internal Assessment: It is defined as the assessment of the learners on the basis of internal evaluation as envisaged in the Credit & Choice based system by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment: It is defined as the assessment of the learners on the basis of Performance in the semester end Theory/ written/ Practical examination.

A. Theory - Internal assessment 25% **25 marks**

Sr. No	Evaluation type	Marks
1.	- Class Test (multiple choice questions / objective)	15
2.	- Mini Project/Case study or Assignment	05
3.	- Active participation in routine class activity - Overall conduct as a responsible student, with respect to good behaviour, leadership qualities, interpersonal skills etc.	05

B. Theory - External examination - 75% **75 marks**

Semester End Theory Assessment

Duration - Each paper shall be of 2.5 hours' duration.

1. Theory question paper pattern: -
    a. There shall be Five compulsory questions, one based on each unit
    b. Each question shall carry 15 marks

c. Each question shall be subdivided into six sub questions a, b, c,d,e,and f with 5 marks each out of which student will have to attempt any three

Question no.	Details	Marks
Q1.	(Based on Unit 1) Attempt any three out of Six:	15
Q2.	(Based on Unit 2) Attempt any three out of Six:	15
Q3.	(Based on Unit 3) Attempt any three out of Six:	15
Q4.	(Based on Unit 4) Attempt any three out of Six:	15
Q5.	(Based on Unit 5) Attempt any three out of Six:	15

C. Semester End Practical Assessment 50 marks

	Based on Each Subject	**50 Marks**
A	Practical Question 1	20
B	Practical Question 2	20
C	Viva	05
D	Journal	05
	OR	
A	Practical Question 1	40
B	Viva	05
C	Journal	05

PRACTICAL BOOK/JOURNAL

The students are required to present a duly certified journal for appearing at the practical examination, failing which they will not be allowed to appear for the examination.

In case of loss of Journal and/ or Report, a Lost Certificate should be obtained from Head/ Coordinator / In-charge of the department; failing which the student will not be allowed to appear for the practical examination.