The Kelkar Education Trust's

V G Vaze College of Arts, Science and Commerce

(Autonomous)

Syllabus for S.Y.B.Sc. – I.T.

(June 2023 Onwards)

Programme: B.Sc.

Subject : Information Technology

Semester III & IV

Semester III			
Course Code	**Course Type**	**Course Title**	**Credits**
SIT301	Skill Enhancement Course	Java Programming	2
SIT302	Core Subject	Computer Networks	2
SIT303	Core Subject	Operating System	2
SIT304	Skill Enhancement Course	Python Programming	2
SIT305	Core Subject	Embedded System	2
SITP301	Skill Enhancement Course Practical	Java Programming Practical	2
SITP302	Core Subject Practical	Computer Networks Practical	2
SITP303	Core Subject Practical	Operating System Practical	2
SITP304	Skill Enhancement CoursePractical	Python Programming Practical	2
SITP305	Core Subject Practical	Embedded System Practical	2
		Total Credits	**20**

Semester – IV			
Course Code	**Course Type**	**Course Title**	**Credits**
SIT401	Skill Enhancement Course	Internet of Things	2
SIT402	Skill Enhancement Course	Advanced Web Programming	2
SIT403	Core Subject	Data Structures	2
SIT404	Core Subject	Software Engineering	2
SIT405	Core Subject	Computer Oriented Statistical and Mathematical Techniques	2
SITP401	Skill Enhancement Course Practical	Internet of Things Practical	2
SITP402	Skill Enhancement Course Practical	Advanced Web Programming Practical	2
SITP403	Core Subject Practical	Data Structures Practical	2
SITP404	Core Subject Practical	Software Engineering Practical	2
SITP405	Core Subject Practical	Computer Oriented Statistical and Mathematical Techniques Practical	2
		Total Credits	**20**

SEMESTER III

B. Sc. (Information Technology)		Semester – III	
Course Name: Java Programming		Course Code: SIT301	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner understand and use
1. Basics of JAVA
2. Control flow, looping statementsand Classesin JAVA.
3. Concept of inheritance and packages.
4. Concept of Enumeration, Arrays, Multithreading, Exception and Byte stream.
5. Concept of event handling, abstract window toolkit and layouts.

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, variablename **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, conditionaloperator.	12
II	**Control Flow Statements, Iteration Statements (Loopings), Jump statements, Enumerations, Arrays:** Two Dimensional Arrays, Multi-DimensionalArrays, Vectors, Adding Elements To A Vector, Accessing Vector Elements, Searching For Elements In A Vector, Working With The Size of The Vector. **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.	12

III	**Inheritance:** Inheritance basics, Types of Inheritance, Derived Class Objects, Inheritance and Access Control, Default Base Class Constructors, this and super keywords, Method overriding, Dynamic method dispatch. Abstract Classes, Abstract Methods,Using final with inheritance Defining An Interface, Implementing Interfaces, Classes V/s Interfaces, Interface can be extended. **Packages:** Creating Packages, Default Package, Importing Packages, Using A Package.	**12**
IV	**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. **Event Handling:** Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes and inner classes.	**12**
V	**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: Introduction to Swing, Difference between AWT and Swing, Swing **Component**s: JButton, JLabel, JTextFiled, JTextArea, JCheckBox, JRadioButton, JMenuBar, JMenu, JMenuItem.	**12**

Course Outcome	
Learner will be able to	
CO1	Implement objectoriented programming concepts in Java.
CO2	Create and use package and interfaces in a Java.
CO3	Develop Java application using graphical user interface.
CO4	Develop windows application using applets.
CO5	Handle events and exception.

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

B. Sc. (Information Technology)		Semester –III	
Course Name: Core Java Practical		Course Code: SITP301	
Periods per week 1 Period is 50 minutes	Lectures per week	3	
		Hours	Marks
Evaluation System	Practical Examination	2	50

List of Practical	
1.	**Java Basics**
a.	Write a Java program that takes a number as input and prints its multiplication table.
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.	WriteaJavaprogramtoconvertadecimalnumbertobinarynumberandviceversa.
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 ofan 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 that demonstrates the use of constructor and destructor.
b.	Write a java program to demonstrate the implementation of abstract class.
5.	**Inheritance and Interface**
a.	Write a java program to implement all types of inheritance.
b.	Write a java program to implement method overriding also show it with dynamic method dispatch.
c.	Write a java program to implement multiple inheritance using interface.
d.	Write a Java program which shows the implementation of interface extending other interface.

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 AWT 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 AWT application that contains the interface to add student information and display the same.
b.	Design a calculator based on AWT application.
c.	Design an AWT application to generate result marks sheet.
11.	Introduction to NetBeans and Eclipse.
a.	Design an Swing application which shows the implementation of swing components using NetBeans/Eclipse.
b.	Design an AWT application which shows the implementation of AWT components using NetBeans/Eclipse.

B. Sc. (Information Technology)		Semester – III	
Course Name: Computer Networks		Course Code: SIT302	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make the learner
1. Aware of Understanding of the fundamental concepts of computer networking as well as will understand functionalities and various concepts of physical layer.
2. Aware of Bandwidth utilization, transmission media and switching concept as well as will understand functionalities and various concepts ofData link layer.
3. Familiarize with basic taxonomy and terminology of computer networking area.
4. Will understand functionalities and various concepts ofNetwork Layer also become aware of advanced networking concepts and implement the same using CISCO packet tracer.
5. Will understand functionalities and various concepts of Transport Layer.

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.	12
II	**Bandwidth Utilization: Multiplexing andSpectrumSpreading:** 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.	12

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.**	**12**
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.	**12**
V	**Introduction to the Transport Layer**:Introduction,Transpor tlayer protocols (Simple protocol, Stop-and-wait protocol, Go-Back-n protocol, Selective repeat protocol, Bidirectional protocols),Transport layer services, User datagram protocol, Transmission control protocol, **Standard Client Server Protocols:**World wide-web and HTTP, FTP, Electronic mail, Telnet, Secured Shell, Domain namesystem.	**12**

Course Outcome	
Learner will be able to	
CO1	Understand and explain Data Communications System and its components.
CO2	Identify the different types of network topologies and protocols.
CO3	Identify the different types of network devices and their functions within a network.
CO4	Understand and building the skills of sub netting and routing mechanisms.
CO5	Motivate the student to become network administrator.

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 Edition	2010
3.	Computer Networks	Andrew Tanenbaum	Pearson	Fifth	2013

B. Sc (Information Technology)		Semester – III	
Course Name: Computer Networks Practical		Course Code: SITP302	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

List of Practical	
1.	**IPv4 Addressing and Subnetting** a. Given an IP address and network mask, determine other information aboutthe IP addresssuchas: • Network address • Network broadcastaddress • Total number of hostbits • Number ofhosts b. Given an IP address and network mask, determine other information aboutthe IP addresssuchas: • The subnet address of thissubnet • The broadcast address of thissubnet • The range of host addresses for thissubnet • The maximum number of subnets for this subnetmask • The number of hosts for eachsubnet • The number of subnetbits • The number of thissubnet
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

B. Sc. (Information Technology)		Semester – III	
Course Name: Operating Systems		Course Code: SIT303	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
Learner should be able
1. To learn the fundamentals of Operating Systems and the mechanisms of OS to handle processes and threads and their communication
2. To learn the mechanisms involved in memory management and file management in contemporary O.S.
3. To learn the handling of input and output as well as deadlock concept in operation system.
4. To gain introductory knowledge on distributed operating system concepts that includes architecture.
5. To learn windows case study or a case study.

Unit	Details	Lectures
I	**Introduction:** What is an operating system? History of operating system, computer hardware, different operating systems, operating system concepts, system calls, operating system structure. **Processes and Threads:** Processes, threads, inter process communication, scheduling	12
II	**Memory Management:** No memory abstraction, memory abstraction: address spaces, virtual memory, page replacement algorithms, design issues or paging systems, implementation issues, segmentation. **File Systems:** files, directories, file system implementation, file-system management and optimization, MS-DOS file system, UNIX V7 file system, CD ROM file system.	12
III	**Input-Output:** Principles of I/O hardware, Principles of I/O software, I/O software layers, disks, clocks, user interfaces: keyboard, mouse, monitor, thin clients, power management, **Deadlocks:** Resources, introduction to deadlocks, the ostrich algorithm, deadlock detection and recovery, deadlock avoidance, deadlock prevention, issues.	12

IV	**Virtualization and Cloud:** History, requirements or virtualization, type 1 and 2 hypervisors, techniques or efficient virtualization, hypervisor micro kernels, memory virtualization, I/O virtualization, Clouds.	**12**
V	**Multiple Processor Systems :** Multiprocessors, multi-computers, distributed systems. **Case Study on Windows:** History of windows through Windows 10, programming windows, system structure, processes and threads in windows, memory management, caching in windows, I/O in windows, Windows NT file system, Windows power management, Security in windows.	**12**

Course Outcome	
Learners should be able to	
CO1	Analyze the structure of OS and basic architectural components involved in OS design.
CO2	Understand the various modules in Operating system like process, memory, files, device and resource management techniques or different types of OS.
CO3	Understand the Mutual exclusion, Deadlock handling methods like Deadlock detection, avoidance.
CO4	Create virtual machine and install Linux / Windows OS.
CO5	Know the basic commands in Linux and Windows OS as well as install and use Windows / Linux Desktop and utilities.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Modern Operating Systems	Andrew S. Tanenbaum, Herbert	Pearson	4th	2014
2.	Operating Systems – Internals and Design Principles	Willaim Stallings	Pearson	8th	2009
3.	Operating System Concepts	Abraham Silberschatz, Peter B. Galvineg Gagne	Wiley	8th	
4.	Operating Systems	Godbole and Kahate	McGraw Hill	3rd	

B. Sc (Information Technology)		Semester – III	
Course Name: Operating Systems Practical		**Course Code: SITP303**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2**	**50**
	Internal	**--**	**--**

List of Practical	
1.	Installation of virtual machine software.
2.	Installation of Linux operating system (RedHat / Ubuntu) on virtual machine.
3.	Installation of Windows operating system on virtial machine.
4.	**Linux commands: Working with Directories:**
a.	pwd, cd, absolute and relative paths, ls, mkdir, rmdir,
b.	file, touch, rm, cp. mv, rename, head, tail, cat, tac, more, less, strings, chmod
5.	**Linux commands: Working with files:**
a.	ps, top, kill, pkill, bg, fg,
b.	grep, locate, find, locate.
c.	date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which.
d.	Compression: tar, gzip.
6.	**Windows (DOS) Commands – 1**
a.	Date, time, prompt, md, cd, rd, path.
b.	Chkdsk, copy, xcopy, format, fidsk, cls, defrag, del, move.
7.	**Windows (DOS) Commands – 2**
a.	Diskcomp, diskcopy, diskpart, doskey, echo
b.	Edit, fc, find, rename, set, type, ver
8.	**Working with Windows Desktop and utilities**
a.	Notepad
b.	Wordpad
c.	Paint
d.	Taskbar
e.	Adjusting display resolution
f.	Using the browsers
g.	Configuring simple networking
h.	Creating users and shares
9.	**Working with Linux Desktop and utilities**
a.	The vi editor.
b.	Graphics
c.	Terminal

d.	Adjusting display resolution
e.	Using the browsers
f.	Coniguring simple networking
g.	Creating users and shares
10.	**Installing utility software on Linux and Windows**

B. Sc. (Information Technology)		Semester – III	
Course Name: Python Programming		**Course Code: SIT304**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal	**--**	**40**

Course Objective
To let the learner
1. Acquire programming skills in core Python.
2. Understand and implement the concept of functions and strings in Python.
3. Understand and implement the concept of Lists, Tuples and Dictionaries, Files and Exception handling.
4. Understand and implement the concepts of Regular expression, Classes and Objects, Multithreading Programming and modules in python.
5. Understand and implement the concepts of Graphical User Interface design and Database(MySQL) programming 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	**12**
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, CheckingTypes	**12**

	Strings: A String Is a Sequence, Traversal with a for Loop, String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, The in Operator, String Comparison, String Operations.	
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, Exceptionwith Arguments, User-defined Exceptions	**12**
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	**12**
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, Messagebox. 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 MySQLdatabase.	**12**

Course Outcome	
Learner will be able to	
CO1	Install and use Python for simple programming tasks.
CO2	Extend the functionality of Python by using add-on packages.
CO3	Develop database (MySQL) application in Python.
CO4	Create GUI based application using python
CO5	Develop a project / application using python.

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, JenniferCampbell, 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

B. Sc. (Information Technology)		Semester – III	
Course Name: Python Programming Practical		Course Code: SITP304	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

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 outa 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). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. 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 getter methods 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 ableto 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 squareAreafrom 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, JenniferCampbell, Paul Gries	SPD	1st	2014

B. Sc. (Information Technology)		Semester – III	
Course Name: Introduction to Embedded Systems		Course Code: SIT305	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make the learner
1. Obtain knowledge about Embedded systems, basic working of a microcontroller system.
2. Obtainthe knowledge of types of embedded systems, embedded hardware and peripherals.
3. Obtainknowledge of 8051 controller and 8051 programming in C language.
4. Understand design and programming of Embedded systems using 8051 controller.
5 Understandthe concept of real time operating system. .

Unit	Details	Lectures
I	**Introduction:** Embedded Systems and general purpose computer systems, history, classifications, applications and purpose ofembedded 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.	12
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, TimerDriver - Watchdog Timers.	12
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.	12

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.	**12**
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, disassemble / de-compiler, simulator, emulator and debugging, embedded product development life-cycle, trends in embedded industry.	**12**

Course Outcome	
Learner will be able to	
CO1	Understand hardware and software required for developing Embedded system application and make a choice for it.
CO2	To design Embedded system.
CO3	Write a basic program of 8051 microcontroller.
CO4	Burn the program on 8051 microcontroller.
CO5	Motivate to develop embedded systems.

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 8051 Microcontroller and Embedded Systems	Muhammad Ali Mazidi	Pearson	Second	2011
4.	Embedded Systems	Rajkamal	Tata Mcgraw-Hill		

B. Sc. (Information Technology)		Semester – III	
Course Name: Introduction to Embedded Systems Practical		Course Code: SITP305	
Periods per week 1 Period is 50 minutes	Lectures per week	3	
		Hours	Marks
Evaluation System	Practical Examination	2	50

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 LEDs.
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.

SEMESTER IV

B. Sc. (Information Technology)		Semester – IV	
Course Name: Internet of Things		Course Code: SIT401	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner aware and use
1. Basics, design principles for connected devices
2. Various types of prototyping, Prototyping embedded devices.
3. Prototyping physical designs and online components.
4. Techniques for writing the embedded code.
5. Business model and ethics.

Unit	Details	Lectures
I	**The Internet of Things: An Overview**: The Flavour of the Internetof Things, The "Internet" of "Things", The Technology of the Internet of Things, EnchantedObjects,Who is Making the Internet of Things? **Design Principles for Connected Devices:** Calm and Ambient Technology, Magic as Metaphor, Privacy, Keeping Secrets, Whose Data Is It Anyway? Web Thinking for Connected Devices, Small Pieces, Loosely Joined, First-Class Citizens on The Internet, Graceful Degradation, Affordances. **Internet Principles:** Internet Communications: An Overview, IP, TCP, The IP Protocol Suite (TCP/IP), UDP, IP Addresses, DNS, Static IP AddressAssignment,Dynamic IP Address Assignment, IPv6, MAC Addresses, TCP andUDP Ports, An Example: HTTP Ports, Other Common Ports, Application Layer Protocols, HTTP, HTTPS: Encrypted HTTP, Other Application Layer Protocols.	12

II	**Thinking About Prototyping:** Sketching, Familiarity, Costs versus Ease of Prototyping, Prototypes and Production, Changing Embedded Platform, Physical Prototypes and Mass Personalisation, climbing into the Cloud, Open Source versus Closed Source, Why Closed? Why Open? Mixing Open and Closed Source, Closed Source for Mass Market Projects, Tapping into the Community. **Prototyping Embedded Devices:** Electronics, Sensors, Actuators, Scaling Up the Electronics, Embedded Computing Basics, Microcontrollers, System-on-Chips, Choosing Your Platform, Arduino, developing on the Arduino, Some Notes on the Hardware, Openness, Raspberry Pi, Cases and Extension Boards, Developing on the Raspberry Pi, Some Notes on the Hardware, Openness.	**12**

Course Outcome	
Learner will be able to	
CO1	Apply the concepts of IOT.
CO2	Write embedded code for IOT based applications.
CO3	Analyse and use data received from sensors.
CO4	Apply IOT to different applications.
CO5	Motivate to develop IOT based applications / projects.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Designing the Internet ofThings	Adrian McEwen, Hakim Cassimally	WILEY	First	2014
2.	Internet of Things – Architecture and Design	Raj Kamal	McGraw Hill	First	2017
3.	Getting Started with the Internet of Things	CunoPfister	O"Reilly	Sixth	2018
4.	Getting Started with Raspberry Pi	Matt Richardson and Shawn Wallace	SPD	Third	2016

B. Sc. (Information Technology)		Semester – IV	
Course Name: Internet of Things Practical		Course Code: SITP401	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

Practical No	Details
	Starting Raspbian OS, Familiarising with Raspberry Pi Components andinterface, Connecting to ethernet, Monitor, USB.
1	Displaying different LED patterns with Raspberry Pi.
2	Displaying Time over 4-Digit 7-Segment Display using Raspberry Pi
3	Raspberry Pi Based Oscilloscope
4	Controlling Raspberry Pi with, Buzzer sensor.
5	Setting up Wireless Access Point using Raspberry Pi
6	Fingerprint Sensor interfacing with Raspberry Pi
7	Raspberry Pi GPS Module Interfacing
8	IoT based Web Controlled Home Automation using Raspberry Pi
9	Visitor Monitoring with Raspberry Pi and Pi Camera
10	Interfacing Raspberry Pi with RFID.
11	Building Google Assistant with Raspberry Pi.
12	Installing Windows 10 IoT Core on Raspberry Pi

B. Sc. (Information Technology)		Semester – IV	
Course Name: Advanced Web Programming		Course Code: SIT402	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner aware of
1. Basics of .NET framework and C# fundamentals.
2. Web form fundamentals and various form controls used in website development using C#.
3. Error handling and state management, master page development in C#.
4. ADO.NET fundamentals, data binding and data controls.
5. XML, Security fundamental and ASP.NET AJAX basics.

Unit	Details	Lectures
I	**Introducing .NET**: The .NET Framework, C#, VB, and the .NET Languages, The Common Language Runtime, The .NET Class Library. **The C# Language**: C# Language Basics, Variables and Data Types, Variable Operations, Object-Based Manipulation, Conditional Logic, Loops, Methods. **Types, Objects, and Namespaces**: The Basics About Classes, Building a Basic Class, Value Types and Reference Types, Understanding Namespaces and Assemblies, Advanced Class Programming.	12
II	**Web Form Fundamentals**: Writing Code, Using the Code-Behind Class, Adding Event Handlers, Understanding the Anatomy of an ASP.NET Application, Introducing Server Controls, Using the Page Class, Using Application Events, Configuring an ASP.NET Application. **Form Controls**: Stepping Up to Web Controls, Web Control Classes, List Controls, Table Controls, Web Control Events and AutoPostBack, Validation, Understanding Validation, Using the Validation Controls, Rich Controls, The Calendar, The AdRotator, Pages with Multiple Views, User Controls and Graphics, User Controls, Dynamic Graphics, The Chart Control, Website Navigation: Site Maps, URL Mapping and Routing, The SiteMapPath Control, The TreeView Control, The MenuControl.	12

III	**Error Handling, Logging, and Tracing**: Avoiding Common Errors, Understanding Exception Handling, Handling Exceptions, Throwing Your Own Exceptions, Using Page Tracing **State Management**: Understanding the Problem of State, Using View State, Transferring Information Between Pages, Using Cookies, Managing Session State, Configuring Session State, Using Application State, Comparing State ManagementOptions **Styles, Themes, and Master Pages**: Styles, Master Page Basics, Advanced Master Pages,	**12**
IV	**ADO.NET Fundamentals**: Understanding Databases, Configuring Your Database, Understanding SQL Basics, Understanding the Data Provider Model, Using Direct Data Access, Using Disconnected Data Access. **Data Binding**: Introducing Data Binding, Using Single-Value Data Binding, Using Repeated-Value Data Binding, Working with Data Source Controls, **The Data Controls**: The GridView, Formatting the GridView, selecting a GridView Row, Editing with the GridView, Sorting and Paging the GridView, Using GridView Templates, The DetailsView and FormView	**12**
V	**XML**: XML Explained, The XML Classes, XML Validation, XML Display and Transforms. **Security Fundamentals**: Understanding Security Requirements, Authentication and Authorization, Forms Authentication, Windows Authentication. **ASP.NET AJAX**: Understanding Ajax, Using Partial Refreshes, Using Progress Notification, Implementing Timed Refreshes, Working with the ASP.NET AJAX ControlToolkit.	**12**

Course Outcome	
Learner will be able to	
CO1	Design a responsive web site and demonstrate Rich Internet Application.
CO2	Implement interactive and dynamic web page(s) using HTML,CSS,AJAX, C# with .NET concepts.
CO3	Develop database oriented web application.
CO4	Build dynamic web site using server side .NET technology and Database connectivity.
CO5	Motivate to work as website developer.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Beginning ASP.NET 4.5 in C#	Matthew MacDonald	Apress		2012
2.	C# 2015	Anne Bohem and Joel Murach	Murach	Third	2016
3.	Murach"s ASP.NET 4.6 Web Programming in C#2015	Mary Delamater and Anne Bohem	SPD	Sixth	2016
4.	ASP.NET4.0 programming	J. Kanjilal	Tata McGraw-Hill		2011
5.	Programming ASP.NET	D.Esposito	Microsoft Press (Dreamtech)		2011
6.	Beginning Visual C# 2010	K. Watson, C. Nagel, J.H Padderson, J.D. Reid, M.Skinner	Wrox (Wiley)		2010

B. Sc. (Information Technology)		Semester – IV	
Course Name: Advanced Web Programming Practical		Course Code: SITP402	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

List of Practical	
1.	**Working with basic C# and ASP .NET**
a.	Create an application that obtains four int values from the user and displays the product.
b.	Create an application to demonstrate string operations.
c.	Create an application that receives the (Student Id, Student Name, Course Name, Date of Birth) information from a set of students. The application should also display the information of all the students once the data entered.
d.	Create an application to demonstrate following operations i. GenerateFibonacciseries. ii. Test for prime numbers. iii. Testforvowels. iv. Use of foreach loop witharrays v. Reverse a number and find sum of digits of a number.
2.	**Working with Object Oriented C# and ASP .NET**
a.	Create simple application to perform following operations i. FindingfactorialValue ii. Money Conversion iii.QuadraticEquation iv. TemperatureConversion
b.	Create simple application to demonstrate use of following concepts i. FunctionOverloading ii. Inheritance (alltypes) iii.Constructoroverloading iv. Interfaces
c.	Create simple application to demonstrate use of following concepts i. Using Delegatesand events ii. Exceptionhandling
3.	**Working with Web Forms and Controls**
a.	Create a simple web page with various sever controls to demonstrate setting and use of their properties. (Example :AutoPostBack)
b.	Demonstrate the use of Calendar control to perform following operations. a) Display messages in a calendar control b) Display vacation in a calendar control c) Selected day in a calendar control using style d) Difference between two calendar dates
c.	Demonstrate the use of Treeview control perform following operations.
	a) Tree view control and data list b) Tree view operations
4.	**Working with Form Controls**

a.	Create a Registration form to demonstrate use of various Validation controls.
b.	Create Web Form to demonstrate use of Adrotator Control.
c.	Create Web Form to demonstrate use User Controls.
5.	**Working with Navigation, Beautification and Master page.**
a.	Create Web Form to demonstrate use of Website Navigation controls and Site Map.
b.	Create a web application to demonstrate use of Master Page with applying Styles and Themes for page beautification.
c.	Create a web application to demonstrate various states of ASP.NET Pages.
6.	**Working with Database**
a.	Create a web application bind data in a multiline textbox by querying in another textbox.
b.	Create a web application to display records by using database.
c.	Demonstrate the use of Datalist link control.
7.	**Working with Database**
a.	Create a web application to display Data binding using dropdown list control.
b.	Create a web application for to display the phone no of an author using database.
c.	Create a web application for inserting and deleting record from a database. (Using Execute-Non Query).
8.	**Working with data controls**
a.	Create a web application to demonstrate various uses and properties of SqlDataSource.
b.	Create a web application to demonstrate data binding using DetailsView and FormView Control.
c.	Create a web application to display Using Disconnected Data Access and Databinding using GridView.
9.	**Working with GridView control**
a.	Create a web application to demonstrate use of GridView control template and GridView hyperlink.
b.	Create a web application to demonstrate use of GridView button column and GridView events.
c.	Create a web application to demonstrate GridView paging and Creating own table format using GridView.
10.	**Working with AJAX and XML**
a.	Create a web application to demonstrate reading and writing operation with XML.
b.	Create a web application to demonstrate Form Security and Windows Security with proper Authentication and Authorization properties.
c.	Create a web application to demonstrate use of various Ajax controls.
11.	**Programs to create and use DLL**

B. Sc. (Information Technology)		Semester – IV	
Course Name: Data Structures		Course Code: SIT403	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner understand and use
1. Basics concepts of algorithms and data structure as well asarrays as data structure as well as compute the complexity of different algorithm.
2. Concept of linked list and different types of linked listsas well as implementation.
3. Concept of stack and queueas well as its implementation in different ways.
4. Concept of Various sorting and searching techniquesas well as trees and advanced tree structure and its implementation.
5 Concept of hashing techniques and graphsas well as its implementation. .

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.	12

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 wayLinked List, Searching in a Two way linked List, Insertion of anelement in Two way Linked List, Deleting a node from Two way Linked List, Header Linked List, Applications of the Linked list,RepresentationofPolynomials,StorageofSparseArrays,Implementing other Data Structures.	**12**
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.	**12**
IV	**Sorting and Searching Techniques :** Bubble, Selection, Insertion, Merge Sort. Searching: Sequential, 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.	**12**
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.	**12**

Course Outcome	
Learner will be able to	
CO1	Select appropriate data structure for a specific problem.
CO2	Implement linear and Non-linear data structure.
CO3	Implement appropriate searching / sorting technique for given problem.
CO4	Determine the complexity of given algorithm.
CO5	Implement different operations on various data structures.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A Simplified Approach toData 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 inC	Weiss, Mark Allen	Addison Wesley	1st	2006

B. Sc. (Information Technology)		Semester – IV	
Course Name: Data Structures Practical		Course Code: SITP403	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

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. [MenuDriven]
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

B. Sc. (Information Technology)		Semester – IV	
Course Name: Software Engineering		**Course Code: SIT404**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal	**--**	**40**

Course Objective
To make learner understand and use
1. Concept of Software engineering, types of software requirements, software development process models.
2. Socio Technical and critical system along with requirement engineering process and different system models used for documentation.
3. Architectural and user interface design for system, project management and quality management
4. Concept of verification and validation, software testing, cost estimation.
5. Concept of serviceoriented software engineering, software reuse and distributed software engineering.

Unit	Details	Lectures
I	**Introduction:** What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenanceetc. **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.** WaterfallModel, Prototyping, IterativeDevelopment, Rational UnifiedProcess, The RADModel, Time boxingModel. **Agile software development**: Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.	**12**

II	**Socio-technical system:**Essential characteristics of socio technical systems, Emergent System Properties, Systems Engineering, Components of system 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, Requirementselicitation and analysis, Requirements Validations, Requirements Management. **System Models:** Models and its types, Context Models, Behavioural Models, Data Models, Object Models, Structured Methods.	**12**
III	**Architectural Design:** Architectural Design Decisions, System Organisation, 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.	**12**
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	**12**
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	**12**

Course Outcome	
Learner will be able to	
CO1	Apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment.
CO2	Work in one or more significant application domains
CO3	Describe the role of SDLC in project development
CO4	Generate project schedule.
CO5	To use the techniques and tools necessary for engineering practice.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Software Engineering, edition,	Ian Somerville	Pearson Education.	Ninth	
2.	Software Engineering	Pankaj Jalote	Narosa Publication		
3.	Software engineering, a practitioner's Approach	Roger Pressman	Tata Mcgraw-hill	Seventh	
4.	Software Engineering principles and practice	WS Jawadekar	Tata Mcgraw-hill		
5.	Software Engineering- A Concise Study	S.A Kelkar	PHI India.		
6.	Software Engineering Concept and Applications	SubhajitDatta	Oxford Higher Education		
7.	Software Design	D.Budgen	Pearson education	2nd	
8.	Software Engineering	KL James	PHI	EEE	2009

B. Sc. (Information Technology)		Semester – IV	
Course Name: Software Engineering		Course Code: SITP404	
Periods per week 1 Period is 50 minutes	Lectures per week	3	
		Hours	Marks
Evaluation System	Practical Examination	2	50

List of Practical (To be executed using Star UML 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.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Object - Oriented Modeling and Design	Michael Blaha, James Rumbaugh	Pearson		2011

The Kelkar Education Trust's

V G Vaze College of Arts, Science and Commerce

(Autonomous)

B. Sc. (Information Technology)		Semester – IV	
Course Name: Computer Oriented Statistical and Mathematical Techniques		Course Code: SIT405	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner understand and use
1. Concept of Matrix, complex numbers.
2. Concept of various measures of Central tendency(Mean, Median Mode), Dispersion(Range , Standard deviation, variance).
3. Concept of moment, skewness and Kurtosis along with elementary probability and sampling theory.
4. Statistical Estimation and Decision theory and Basics of R.
5. The sampling theory and testing of hypothesis and making inferences and Chi- Square test
6. Concept of Curve Fitting and the Method of Least Squares as well as Correlation Theory.

Unit	Details	Lectures
I	**Matrices:** Inverse of a matrix, Properties of matrices, Elementary Transformation , Rank of Matrix , Echelon or Normal Matrix ,Inverse of matrix, Linear equations, Linear dependence and linear independence of vectors, Linear transformation, Characteristics roots and characteristics vectors, Properties of characteristic vectors, Caley- Hamilton Theorem, Similarity of matrices, Reduction of matrix to adiagonal matrix which has elements as characteristics values. **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 andMeasures of Central Tendency.	12

II	**The Standard Deviation and Other Measures of Dispersion**: Dispersion, or Variation,The Range, The Mean Deviation, The Semi- Interquartile Range,The 10–90 Percentile Range, The StandardDeviation, The Variance, Short Methods for Computing the Standard Deviation, Properties of the Standard Deviation, Empirical Relations Between Measures of Dispersion, Absolute and Relative Dispersion; Coefficient of Variation, Standardized Variable; Standard Scores, Software and Measures ofDispersion.**Moments, Skewness, and Kurtosis :**Moments , Moments for Grouped Data ,Relations Between Moments , Computation of Moments for Grouped Data, Moments in Dimensionless Form, Skewness, Kurtosis, Population Moments, Skewness, and Kurtosis, Software Computation of Skewness andKurtosis. **Elementary Probability Theory**: Definitions of Probability, Conditional Probability; Independent and Dependent Events, Mutually Exclusive Events, Combinations and Permutation.	**12**
III	**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**.** **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,Operating-Characteristic Curves; the Power of a Test, p- Values for Hypotheses Tests, Control Charts, Tests Involving Sample Differences, Tests Involving Binomial Distributions.	**12**
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.	**12**

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.	**12**

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A text book of AppliedMathematics Vol I	P. N. Wartikar and J. N. Wartikar	Pune VidyathiGraha		
2.	Statistics	Murray R. Spiegel, Larry J. Stephens.	Mcgraw –Hill Iternational	Fourth	
3.	A Practical Approachusing R	R.B. Patil, H.J. Dand and R. Bhavsar	SPD	1st	2017
4.	Fundamental of Mathematical Statistics	S.C. GUPTAand V.K. KAPOOR	Sultan Chand AndSons	Eleventh revised	2011
5.	Mathematical Statistics	J.N. Kapur and H.C. Saxena	S. Chand	Twentieth revised	2005

B. Sc. (Information Technology)		Semester – IV	
Course Name: Computer Oriented Statistical Techniques Practical		Course Code: SITP405	
Periods per week 1 Period is 50 minutes	Lectures per week	3	
		Hours	Marks
Evaluation System	Practical Examination	2	50

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, transposeand 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