SOPHIA COLLEGE FOR WOMEN (AUTONOMOUS)

Affiliated to

UNIVERSITY OF MUMBAI

Programme: Information Technology

Programme Code: SBTTEC

S.Y.B.Sc. I.T.
2021 - 22

(Choice Based Credit System with effect from the year 2018-19)

Programme Outline: SYBScIT (SEMESTER III)

SEMESTER – 3			
COURSE CODE	**UNIT NO**	**COURSE TITLE**	**CREDITS**
SBTTEC301		**PYTHON PROGRAMMING**	2
	1	Introduction, variables, expressions, conditional statements, looping and control statements	
	2	Functions, strings	
	3	lists, tuples, dictionaries and files Exceptions	
	4	Regular expressions, classes and objects, multithreaded programming, modules	
	5	GUI Forms, Widgets, Layout, Look and feel, MySQL database	
SBTTEC302		**DATA STRUCTURES**	2
	1	Introduction and array	
	2	Linked list	
	3	Stack and queue	
	4	Sorting, searching, tree and advance tree	
	5	Hashing and graph	
SBTTEC303		**COMPUTER NETWORKS**	2
	1	Introduction, network models, introduction to physical layer, digital and analog transmission	
	2	Bandwidth utilization, multiplexing, transmission media, switching, introduction to data link layer	
	3	Data link, media access control, wireless lan and virtual lan	
	4	Network layer, unicast routing and Next generation IP	
	5	Introduction to the Transport Layer, Standard Client0Server Protocols	
SBTTEC304		**DATABASE MANAGEMENT SYSTEMS**	2

	1	Introduction to Databases and Transactions, Data Models, Database Design, ER Diagram andUnified Modeling Language	2
	2	Relational database model: Relational Algebra and Calculus	
	3	Constraints, Views and SQL	
	4	Transaction management and Concurrency	
	5	PL-SQL	
SBTTEC305		**Computer Oriented Statistical Techniques**	2
	1	The Mean, Median, Mode, and Other Measures of Central Tendency, The Standard Deviation and Other Measures of Dispersion, Introduction to R	
	2	Moments, Skewness, and Kurtosis, Elementary Probability Theory, Elementary Sampling Theory	
	3	Statistical Estimation Theory, Statistical Decision Theory, Statistics in R	
	4	Small Sampling Theory, The Chi-Square Test	
	5	Curve Fitting and the Method of Least Squares, Correlation Theory	
SBTTECP301		PYTHON PROGRAMMING PRACTICAL	2
SBTTECP302		DATA STRUCTURES PRACTICAL	2
SBTTECP303		COMPUTER NETWORKS PRACTICAL	2
SBTTECP304		DATABASE MANAGEMENT SYSTEMS PRACTICAL	2
SBTTECP305		COMPUTER ORIENTED STATISTICAL TECHNIQUES PRACTICAL	2
		Total Credits	**20**

Programme Outline: SYBScIT (SEMESTER IV)

SEMESTER – IV			
COURSE CODE	**UNIT NO**	**COURSE TITLE**	**CREDITS**
SBTTEC401		**CORE JAVA**	2
	1	Introduction and Data types	
	2	Control Flow Statements, Iterations, Classes	
	3	Inheritance and Packages	
	4	Enumerations, Arrays, Multithreading, Exceptions and Byte streams	
	5	Event Handling, Abstract Window Toolkit, Layouts	
SBTTEC402		**COMPUTER FORENSICS**	2
	1	Introduction to Cyber Crimes, Computer Forensics and Investigations as a Profession	
	2	Understanding Forensic Investigations, Crime Scene Investigations	
	3	The Investigator's Office and Laboratory, Data Acquisitions	
	4	Processing Crime and Incident Scenes, Computer Forensics Tools	
	5	Cell Phone and Mobile Device Forensics, Internet Forensics, Investigation, Evidence presentation and Legal aspects of Digital Forensics	
SBTTEC403		**ARTIFICIAL INTELLIGENCE**	2
	1	Introduction, Intelligent Agents	

	2	Solving Problems by Searching, Beyond Classical Search	
	3	Adversarial Search, Logical Agents	
	4	First Order Logic, Inference in First Order Logic	
	5	Planning, Knowledge Representation	
SBTTEC404		**IT SERVICE MANAGEMENT**	2
	1	IT Service Management, Service Strategy Principles, Service Strategy	
	2	Service Design Service Design Principles Service Design Processes	
	3	Service Transition Service Transition Principles Service Transition Processes	
	4	Service Operation Service Operation Principles Service Operation Processes	
	5	Continual Service Improvement(CSI) Principles CSI Methods and Techniques Organising for CSI Implementing CSI	
SBTTEC405		**COMPUTER GRAPHICS AND ANIMATION**	2
	1	Introduction to Computer Graphics, Scan conversion	
	2	Two-Dimensional Transformations and Three-Dimensional Transformations	
	3	Viewing in 3D , Light, color	
	4	Visible-Surface Determination, Plane Curves and Surfaces	
	5	Computer Animation and Image Manipulation and Storage	
SBTTECP401	1	CORE JAVA PRACTICAL	2

SBTTECP402	2	COMPUTER FORENSICS PRACTICAL	2
SBTTECP403	3	ARTIFICIAL INTELLIGENCE PRACTICAL	2
SBTTECP404	4	ADVANCED MOBILE PROGRAMMING PRACTICAL	2
SBTTECP405	5	COMPUTER GRAPHICS AND ANIMATION PRACTICAL	2
		Total Credits	**20**

Preamble:

Information Technology (IT) refers to the use, development, and management of computer systems, software, and networks to process, store, retrieve, and exchange information. It encompasses a broad range of technologies and practices aimed at solving problems, improving efficiency, and enabling communication within and between organizations and individuals.

In an era marked by rapid digital transformation and technological advancements, our program is designed to equip students with a comprehensive understanding of the foundational and emerging concepts in Information Technology.

Our BSc IT curriculum integrates theoretical knowledge with practical skills, preparing students to tackle real-world challenges and excel in a diverse range of IT careers. Through a combination of rigorous coursework, industry-relevant projects, and learning experiences, we aim to develop well-rounded professionals who are adept at problem-solving and equipped with the tools to drive technological innovation.

PROGRAMME OBJECTIVES:

PO 1	To think analytically and creatively in developing robust, extensible and maintainable technological solutions to simple and complex problems.
PO 2	To work effectively as a part of a team to achieve a common stated goal.
PO 3	To imbibe quality software development practices.
PO 4	To apply their knowledge and skills to be employed and excel in IT professional careers and/or to continue their education in IT and/or related post graduate programmes.
PO 5	To communicate effectively with a range of audiences both technical and non-technical.

PROGRAMME SPECIFIC OUTCOMES:

PSO 1	The Learner will be able to demonstrate a strong understanding of fundamental concepts in information technology including programming, databases, networking, and software engineering principles.
PSO 2	The Learner will be able to apply technical skills in software development, system analysis, and design using contemporary tools and technologies.
PSO 3	The Learner will able to have proficiency in identifying, formulating, and solving IT-related problems using appropriate techniques, algorithms, and methodologies.
PSO 4	The Learner will be able to have understanding of project management principles and methodologies relevant to IT projects, including planning, scheduling, and resource management
PSO 5	The Learner will be able to have effective communication skills, both oral and written, necessary for articulating technical concepts and collaborating in a team environment.

Semester – III		
NAMEOF THE COURSE	PYTHON PROGRAMMING	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC301	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1.	The course aims to train the student to the basic concepts of the Python programming language.
CO 2.	It aims to train the students to understand the concept of conditional statement, loop, nested loop and control statement.
CO 3.	It aims to train the students to understand the concept of function and string.
CO 4.	It aims to train the students to understand the concept List, Tuple and Dictionary in Python.
CO 5.	It aims to train the students to understand the concept Object Oriented Programming Paradigm, Regular Expression and Exception handling.

COURSE LEARNING OUTCOME:

CLO 1.	Read, understand and trace the execution of programs in Python language.
CLO 2.	Implement the concept of control statements, loops, and functions to write a Python program.
CLO 3.	To develop Programs with concept of List, Tuple and Dictionary in Python.
CLO 4.	To develop Programs with the concept of Object-Oriented Programming Paradigm in Python.
CLO 5.	Implement the concept of multithreading and exception handling in Python.

UNIT 1	**Introduction, variables, expressions,conditional statements, looping and control statements (15 LECTURES)**
1.1	**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
1.2	**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.
1.3	**Conditional Statements:** if, if-else, nested if –else
1.4	**Looping**: for, while, nested loops
1.5	**Control statements:** Terminating loops, skipping specific conditions
UNIT 2	**Functions, strings (15 LECTURES)**
2.1	**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
2.2	**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.
UNIT 3	**Lists, Tuples and Dictionaries, Files (15 LECTURES)**
3.1	**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
3.2	**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
3.3	**Files:** Text Files, The File Object Attributes, Directories

3.4	**Exceptions:** Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions
UNIT 4	**Regular expressions, classes and objects, multithreaded programming and modules (15 LECTURES)**
4.1	**Regular Expressions –** Concept of regular expression, various types of regular expressions, using match function.
4.2	**Classes and Objects:** Overview of OOP (Object Oriented Programming), Class Definition, Creating Objects, Instances
4.3	**Multithreaded Programming:** Thread Module, creating a thread, synchronizing threads, multithreaded priority queue
4.4	**Modules:** Importing module, Creating and exploring modules, Math module, Random module, Time module
UNIT 5	**GUI Forms, Widgets, Layout, Look and feel, MySQL database (15 LECTURES)**
5.1	**Creating the GUI Form and Adding Widgets:**
5.2	**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.
5.3	**Layout Management:** Designing GUI applications with proper Layout Management features.
5.4	**Look and Feel Customization:** Enhancing Look and Feel of GUI using different appearances of widgets.
5.5	**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.

REFERENCES:

- Think Python Allen Downey O'Reilly 1st 2012
- Introduction to Problem Solving with Python E. Balagurusamy TMH 1st 2016
- Core Python Programming, Dr. R. Nageshwar Rao, Dreamtech Press 201

NAME OF THE COURSE	PYTHON PROGRAMMING PRACTICAL	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP301	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PERWEEK	3	
TOTAL NUMBER OF LECTURESPER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

List of Practical: (Can be done in any imperative language)	
1.	**Write the program for the following:**
1.1.	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..
1.2.	Enter the number from the user and depending on whether the number is even or odd, print out an appropriate message to the user.
1.3.	Write a program to generate the Fibonacci series.
1.4	Write a function that reverses the user defined value.
1.5	Write a function to check the input value is Armstrong and also write the function for Palindrome.
1.6	Write a recursive function to print the factorial for a given number.
2.	**Write the program for the following:**
2.1.	Write a function that takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise.
2.2.	Define a function that computes the *length* of a given list or string.

2.3.	Define a *procedure* histogram() that takes a list of integers and prints a histogram to thescreen. For example, histogram([4, 9, 7]) should print the following: **** ********* *******
3.	**Write the program for the following:**
3.1.	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 towrite a function to check a sentence to see if it is a pangram or not.
3.2.	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:**
4.1.	Write a program that takes two lists and returns True if they have at least one common member.
4.2.	Write a Python program to print a specified list after removing the 0th, 2nd, 4th and 5th elements.
4.3.	Write a Python program to clone or copy a list
5.	**Write the program for the following:**
5.1.	Write a Python script to sort (ascending and descending) a dictionary by value
	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}
	Write a Python program to sum all the items in a dictionary.
6.	**Write the program for the following:**
6.1.	Write a Python program to read an entire text file.
6.2	Write a Python program to append text to a file and display the text.
6.3	Write a Python program to read last n lines of a file.
7.	**Write the program for the following:**

7.1.	Design a class that store the information of student and display the same
7.2.	Implement the concept of inheritance using python
7.3	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. Write a static method called subtract, which takes two number parameters, b andc, 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 a deleter for manipulating thevalues of x and y.
8.	**Write the program for the following:**
8.1.	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 import your own module like this: import geometry
8.2.	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.
8.3.	Write a program to implement exception handling.

9.	**Write the program for the following:**
9.1	Try to configure the widget with various options like: bg=”red”, family=”times”, size=18
9.2.	Try to change the widget type and configuration options to experiment with other widget types like Message, Button, Entry, Checkbutton, Radiobutton, Scale etc.
10.	**Write the program for the following:**
10.1	Design a simple database application that stores the records and retrieve the same.
10.2	Design a database application to search the specified record from the database.

Semester – III		
NAMEOF THE COURSE	DATA STRUCTURES	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC302	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1.	Allow to assess how the choice of data structures and algorithm designmethods impacts the performance of programs
CO 2.	To provide the knowledge of basic data structures and their implementations.
CO 3.	To understand the concept of Dynamic memory management, data types, algorithms, asymptotic analysis and notation.
CO 4.	To understand the importance of data structures in context of writing efficient programs.
CO 5.	To develop skills to apply appropriate data structures in problem solving.

COURSE LEARNING OUTCOMES:

CLO 1.	Learn the basic types for data structure, implementation and application.
CLO 2.	Know the strength and weakness of different data structures.
CLO 3.	Use the appropriate data structure in context of solution of given problem.
CLO 4.	Develop programming skills which require for solving given problem.
CLO 5.	Ability to estimate the algorithmic complexity of simple, non-recursiveprograms.

Unit 1	**Introduction and array (15 LECTURES)**
1.1	**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
1.2	**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 MultiDimensional Arrays, Sparse Arrays, Sparse Matrix, Memory
Unit 2	**Linked list (15 LECTURES)**
2.1	**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.
Unit 3	**Stack and queue (15 LECTURES)**
3.1	**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.
3.2	**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.
Unit 4	**Sorting, searching, tree and advance**

	tree (15 LECTURES)
4.1	**Sorting and Searching Techniques**: Bubble, Selection, Insertion, Merge Sort. Searching: Sequential, Binary, Indexed Sequential Searches, Binary Search.
4.2	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.
4.3	Advanced Tree Structures: Red Black Tree, Operations Performed on Red Black Tree, AVL Tree, Operations performed on AVL Tree, 23 Tree, B-Tree.
Unit 5	**Hashing and graph (15 LECTURES)**
5.1	**Hashing Techniques:** Hash function, Address calculation techniques, Common hashing functions Collision resolution, Linear probing, Quadratic, Double hashing, Bucket hashing, Deletion and rehashing
5.2	**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, Graph Traversal, Applications of the Graph, Reachability, Shortest Path Problems, Spanning Trees.

REFERENCES:

- Data Structures by Lipschutz, Seymour
- Data Structure and algorithm analysis in C – 2nd Edition by Weiss, Mark Allen
- A simplified approach to Data Structures - 5th Edition by Goyal, Vishal and others

NAME OF THE COURSE	DATA STRUCTURES PRACTICAL	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP302	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	3	
TOTAL NUMBER OF LECTURES PER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

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

4.1.	Write a program to implement the concept of Queue with Insert, Delete, Display and Exit operations.
4.2.	Write a program to implement the concept of Circular Queue
4.3.	Write a program to implement the concept of Deque.
5.	**Implement the following sorting techniques:**
5.1.	Write a program to implement bubble sort.
5.2.	Write a program to implement selection sort.
5.3.	Write a program to implement insertion sort.
6.	**Implement the following data structure techniques:**
6.1	Write a program to implement merge sort.
6.2	Write a program to search the element using sequential search.
6.3	Write a program to search the element using binary search.
7.	**Implement the following data structure techniques:**
7.1	Write a program to create the tree and display the elements.
7.2	Write a program to construct the binary tree.
7.3	Write a program for inorder, postorder and preorder traversal of tree
8.	**Implement the following data structure techniques:**
8.1.	Write a program to insert the element into maximum heap.
8.2.	Write a program to insert the element into minimum heap.
9.	**Implement the following data structure techniques:**
9.1	Write a program to implement the collision technique.
9.2	Write a program to implement the concept of linear probing.
10.	**Implement the following data structure techniques:**
10.1	Write a program to generate the adjacency matrix.
10.2	Write a program for shortest path diagram.

Semester – III		
NAMEOF THE COURSE	COMPUTER NETWORKS	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC303	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1.	To learn to differentiate between topologies, networking devices, OSI and TCP/IP models.
CO 2.	To able to identify and describe various techniques for efficient bandwidth utilization under wired and wireless medium
CO 3.	To distinguish between various wireless network models.
CO 4.	To be able to analyze the different networking protocols and Ip header formats
CO 5.	To be able to explain the different acknowledging schemes used in case data loss

COURSE LEARNING OUTCOMES:

CLO 1.	State the functionality of each layer of OSI model when the data is passed from sender toreceiver
CLO 2.	Compare FDM, TDM and WDM
CLO 3.	Explain the working of cellular telephony
CLO 4.	State the reason why ipv6 is more robust than ipv4
CLO 5.	Describe the difference in TCP and UDP header formats

Unit 1	**Introduction, network models, introduction to physical layer, digitaland analog transmission (15 LECTURES)**
1.1	**Introduction:** Data communications, networks, network types, Internet history, standards and administration.
1.2	**Network Models:** Protocol layering, TCP/IP protocol suite, The OSI model.
1.3	**Introduction to Physical layer:** Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance.
1.4	**Digital and Analog transmission:** Digital-to-digital conversion, analog-to-digital conversion, transmission modes, digital-to-analog conversion, analog-to-analog conversion.
Unit 2	**Bandwidth utilization, multiplexing,transmission media, switching, introduction to data link layer (15 LECTURES)**
2.1	**Bandwidth Utilization: Multiplexing and Spectrum Spreading:** Multiplexing, Spread Spectrum
2.2	**Transmission media:** Guided Media**,** Unguided Media
2.3	**Switching:** Introduction, circuit switched networks, packet switching, structure of a switch.
2.4	**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.
Unit 3	**Data link, media access control,wireless lan and virtual lan (15 LECTURES)**
3.1	**Data Link Control:** DLC services, data link layer protocols, HDLC, Point-to-point protocol.
3.2	**Media Access Control**: Random access, controlled access, channelization, Wired LANs – Ethernet Protocol, standard ethernet, fast ethernet, gigabit ethernet, 10 gigabit ethernet,
3.3	**Wireless LANs:** Introduction, IEEE 802.11 project, Bluetooth, WiMAX, Cellular telephony, Satellite networks.
Unit 4	**Network layer, unicast routing andNext generation IP (15 LECTURES)**

4.1	**Introduction to the Network Layer:** Network layer services, packet switching, network layer performance, IPv4 addressing, forwarding of IP packets, Internet Protocol, ICMPv4, Mobile IP
4.2	**Unicast Routing:** Introduction, routing algorithms, unicast routing protocols.
4.3	**Next generation IP:** IPv6 addressing, IPv6 protocol, ICMPv6 protocol, transition from IPv4 to IPv6.
Unit 5	**Introduction to the Transport Layer,Standard Client0Server Protocols (15 LECTURES)**
5.1	**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 layer services, User datagram protocol, Transmission control protocol,
5.2	**Standard Client Server Protocols:** World wide-web and HTTP, FTP, Electronic mail, Telnet, Secured Shell, Domain name system.

REFERENCES:

- Forouzan, Behrouz A ,Data communication and networking.5th ed
- Tanenbaum,Andrew S.& Wetherall ,David J. , Computer networks 5th ed.
- Forouzan, Behrouz A. Tcp /IP Protocol suite.4th ed.

NAME OF THE COURSE	COMPUTER NETWORKS PRACTICAL	
CLASS	FYBSCIT	
COURSE CODE	SBTTECP303	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	3	
TOTAL NUMBER OF LECTURES PER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

List of Practical	
1.	**IPv4 Addressing and Subnetting**
1.1	Given an IP address and network mask, determine other information about theIP address such as: Network address Network broadcast address Total number of host bits Number of hosts
1.2	Given an IP address and network mask, determine other information about theIP address such 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 bitsThe 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 • SMT • POP3

Semester – III		
NAMEOF THE COURSE	DATABASE MANAGEMENT SYSTEM	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC304	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1.	The main objective of this course is to enable students to learn the fundamental concepts of database management system and design.
CO 2.	To emphasize the importance of normalization in databases. Discuss normalization techniques and relational algebra concepts which helps in understanding queries.
CO 3.	To demonstrate the use of Integrity constraints. Students will be able to understand and write various advanced queries.
CO 4.	Understanding the properties of transaction management and concurrency control methods.
CO 5.	Beginning with PL / SQL and learning Control Structures, Cursors, Procedures, Functions, Exceptions Handling and Packages.

COURSE LEARNING OUTCOMES:

CLO 1.	Explain basic database concepts, data models, Unified Modeling language, schemas and instances. Compare file systems and database management system. Draw entity relationship diagrams using appropriate components.
CLO 2.	Explain the importance of normalization in databases. Discuss normalization techniques and various types of joins. Explain the use of relational algebra

	concepts.
CLO 3.	State and explain the use of Integrity constraints. Write SQL queries involving advanced concepts.
CLO 4.	State and explain the properties of transaction management and concurrency control methods.
CLO 5.	Write PL / SQL programs using various Control Structures, Cursors, Procedures, Functions, Exceptions Handling and Packages.

Unit 1	**Introduction to Databases andTransactions, Data Models, Database Design, ER Diagram andUnified Modeling Language (15 LECTURES)**
1.1	**Introduction to Databases and Transactions** What is database system, purpose of database system, view of data, relational databases, database architecture, transaction management
1.2	**Data Models** The importance of data models, Basic building blocks, Business rules, The evolution of data models, Degrees of data abstraction.
1.3	**Database Design, ER Diagram and Unified Modeling Language** Database design and ER Model: overview, ER Model, Constraints, ER Diagrams, ERD Issues, weak entity sets, Codd's rules, Relational Schemas, Introduction to UML
Unit 2	**Relational database model: Relational Algebra and Calculus (15 LECTURES)**
2.1	**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).
2.2	**Relational Algebra and Calculus:** Relational algebra: introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison.
2.3	**Calculus:** Tuple relational calculus, Domain relational Calculus, calculus vs algebra, computational capabilities

Unit 3	**Constraints, Views and SQL (15 LECTURES)**
3	**Constraints, Views and SQL** Constraints, types of constrains, Integrity constraints, Views: Introduction to views, data independence, security, updates on views, comparison between tables and viewsSQL: data definition, aggregate function, Null Values, nested sub queries, Joined relations. Triggers.
Unit 4	**Transaction management and Concurrency (15 LECTURES)**
4	**Transaction management and Concurrency** Control Transaction management: ACID properties, serializability and concurrency control, Lock based concurrency control (2PL, Deadlocks), Time stamping methods,optimistic methods, database recovery management.
Unit 5	**PL-SQL (15 LECTURES)**
5.1	**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
5.2	Exceptions Handling, Packages, With Clause and Hierarchical Retrieval, Triggers.

REFERENCES:

- Database System and ConceptsA Silberschatz, H Korth, S Sudarshan McGraw-Hill Fifth Edition
- Introduction to Database SystemC.J.DatePearsonPearson 2003
- Database Systems Rob Coronel Cengage Learning Twelfth Edition
- Oracle database 11g PL/SQL programming McLaughlin, Michael

NAME OF THE COURSE	DATABASE MANAGEMENT SYSTEM PRACTICAL	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP304	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	3	
TOTAL NUMBER OF LECTURES PER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

List of Practical: Write the programs for the following	
1.	**SQL Statements – 1**
1.1	Writing Basic SQL SELECT Statements
1.2	Restricting and Sorting Data
1.3	Single-Row Functions
2.	**SQL Statements – 2**
2.1.	Displaying Data from Multiple Tables
2.2.	Aggregating Data Using Group Functions
2.3.	Subqueries
3.	**Manipulating Data**
3.1.	Using INSERT statement
3.2.	Using DELETE statement
3.3	Using UPDATE statement
4.	**Creating and Managing Tables**
4.1.	Creating and Managing Tables
4.2.	Including Constraints
5	**Creating and Managing other database objects**
5.1	Creating Views
5.2	Other Database Objects
5.3	Controlling User Access
6	**Using SET operators, Date/Time Functions, GROUP BY clause (advanced features) and advanced subqueries**
6.1	Using SET Operators
6.2	Datetime Functions
6.3	Enhancements to the GROUP BY Clause
6.4	Advanced Subqueries
7	**PL/SQL Basics**
7.1	Declaring Variables

7.2	Writing Executable Statements
7.3	Interacting with the Oracle Server
7.4	Writing Control Structures
8	**Composite data types, cursors and exceptions.**
8.1	Working with Composite Data Types
8.2	Writing Explicit Cursors
8.3	Handling Exceptions
9	**Procedures and Functions**
9.1	Creating Procedures
9.2	Creating Functions
9.3	Managing Subprograms
9.4	Creating Packages
10	**Creating Database Triggers**
10.1	**SQL Statements – 1**
10.2	Writing Basic SQL SELECT Statements
10.3	Restricting and Sorting Data
10.4	Single-Row Functions
11	**SQL Statements – 2**
11.1	Displaying Data from Multiple Tables
11.2	Aggregating Data Using Group Functions
11.3	Subqueries
12.	**Manipulating Data**
12.1.	Using INSERT statement
12.2.	Using DELETE statement
12.3.	Using UPDATE statement

Semester – III		
NAMEOF THE COURSE	**COMPUTER ORIENTED STATISTICAL TECHNIQUES**	
CLASS	S.Y.B.Sc. IT	
COURSE CODE	SBTTEC305	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBER OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1	Obtain an intuitive and working understanding of numerical methods for the basic problemsof numerical analysis.
CO 2	Formulate and solve linear programming problems and operations with nonlinear expressions
CO 3	Gain experience in the implementation of numerical methods using a computer. Trace errorin these methods and need to analyze and predict it.
CO 4	Provide knowledge of various significant and fundamental concepts to inculcate in the students an adequate understanding of the application of Statistical Methods.
CO 5	Demonstrate the concepts of numerical methods used for different applications

COURSE LEARNING OUTCOMES

CLO 1	To calculate and apply measures of central tendencies and measures of dispersion -- grouped and ungrouped data cases.
CLO 2	To calculate the moments, skewness and kurtosis by various methods.
CLO 3	How to apply discrete and continuous probability distributions to various business problems
CLO 4	Perform Test of Hypothesis as well as calculate confidence interval for a population parameter for single sample and two sample cases. Understand the concept of p-values
CLO 5	Apply simple linear regression and correlation model to real life examples

Unit 1	**The Mean, Median, Mode, and Other Measures of Central Tendency, The Standard Deviation and Other Measures of Dispersion, Introduction to R**
1.1	**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.
1.2	**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 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.
1.3	**Introduction to R:** Basic syntax, data types, variables, operators, control statements, R-functions, R –Vectors, R – lists, R Arrays.
Unit 2	**Moments, Skewness, and Kurtosis, Elementary Probability Theory, Elementary Sampling Theory**
2.1	**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.
2.2	**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, Sterling's Approximation to n!, Relation of Probability to Point Set Theory, Euler or Venn Diagrams and Probability
2.3	**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.
Unit 3	**Statistical Estimation Theory, Statistical Decision Theory, Statistics in R**
3.1	**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.
3.2	**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
3.3	**Statistics in R:** mean, median, mode, Normal Distribution, Binomial Distribution, Frequency Distribution in R.
Unit 4	**Small Sampling Theory, The Chi-Square Test**
4.1	**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.
4.2	**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.
Unit 5	**Curve Fitting and the Method of Least Squares, Correlation Theory**
5.1	**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.
5.2	**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

REFERENCES:

1. A text book of AppliedMathematics Vol I P. N. Wartikarand J. N. Wartikar Pune VidyathiGraha
2. Applied Mathematics II P. N. Wartikar and J. N.Wartikar Pune VidyathiGraha
3. Higher EngineeringMathematics Dr. B. S. Grewal Khanna Publications
4.

NAME OF THE COURSE	**COMPUTER ORIENTED STATISTICAL TECHNIQUES**	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP305	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PERWEEK	3	
TOTAL NUMBER OF LECTURESPER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

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 standarddeviation, 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

SEMESTER IV

Semester – IV		
NAMEOF THE COURSE	CORE JAVA	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC401	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1.	To introduce the basic concepts of Java and its data types.
CO 2.	To gain knowledge about the control flow statement, iterations and classes in Java.
CO 3.	To become familiar with concept of inheritance and packages.
CO 4.	To use enumerations, arrays, multithreading, exceptions and byte streams with ease.
CO 5.	To study concepts of event handling, abstract window toolkit and layouts.

COURSE LEARNING OUTCOMES:

CLO 1.	Use the syntax and semantics of java programming language and basic concepts of OOP.
CLO 2.	Implement the use of a variety of basic control structures including selection and repetition; classes and objects.
CLO 3.	Develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages.
CLO 4.	Apply the concepts of Array, Multithreading and Exception handling to develop efficient and error free codes.
CLO 5.	Design event driven GUI and web related applications.

Unit 1	Introduction and Data types (15 LECTURES)

1.1	**Introduction:** History, architecture and its components, Java Class File, Java RuntimeEnvironment, 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
1.2	**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.
Unit 2	**Control Flow Statements, Iterations,Classes (15 LECTURES)**
2.1	**Control Flow Statements:** The If…Else If…Else Statement, The Switch…Case Statement
2.2	**Iterations:** The While Loop, The Do … While Loop, The For Loop, The Foreach Loop, Labeled Statements, The Break And Continue Statements, The Return Statement
2.3	**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.
Unit 3	**Inheritance and Packages (15 LECTURES)**
3.1	**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, WhatIs 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
3.2	.**Packages:** Creating Packages, Default Package, Importing Packages, Using A Package.
Unit 4	**Enumerations, Arrays, Multithreading, Exceptions and Byte streams (15 LECTURES)**
4.1	**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.
4.2	**Multithreading:** the thread control methods, thread life cycle, the main thread, creating a thread, extending the thread class.
4.3	**Exceptions:** Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause
4.4	**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
Unit 5	**Event Handling, Abstract Window Toolkit, Layouts (15 LECTURES)**
5.1	**Event Handling:** Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes and inner classes.
5.2	**Abstract Window Toolkit:** Window Fundamentals, Component, Container, Panel, Window, Frame, Canvas. Components – Labels, Buttons, Check Boxes, Radio Buttons, Choice Menus, Text Fields,
5.3	Text, Scrolling List, Scrollbars, Panels, Frames **Layouts:** Flow Layout, Grid Layout, Border Layout, Card Layout.

REFERENCES:

- Core Java for beginners, Shah, Sharanam & Shah, Vaishali Shroff Publishers & Distributors, 2010
- Java the complete reference. 9th ed , Schildt, Herbert, McGraw Hill Education (India), 2014

- Core Java: An integrated approach. Covers concepts, programs and interview questions. Rao, R. Nageswara, Dreamtech Press, 2017
- Core Java. Volume.II: Advanced features. 9th ed. , Horstmann, Cay S. & Cornell, Gary Dorling Kindersley (India) 2013

NAME OF THE COURSE	CORE JAVA PRACTICAL	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP401	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	3	
TOTAL NUMBER OF LECTURES PER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

List of Practical: To be implemented using object oriented language	
1.	**Java Basics**
1.1	Write a Java program that takes a number as input and prints its multiplication table upto 10.
1.2	Write a Java program to display the following pattern. ***** **** *** ** *
1.3	Write a Java program to print the area and perimeter of a circle.
2.	**Use of operators**
2.1	Write a Java program to add two binary numbers.
2.2	Write a Java program to convert a decimal number to binary number and vice versa.
2.3	Write a Java program to reverse a string.
3.	**Java Data Types**
3.1	Write a Java program to count the letters, spaces, numbers and other characters of an input string.
3.2	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.

3.3.	Find the smallest and largest element from the array
4.	**Methods and Constructors**
4.1	Designed a class SortData that contains the method asec() and desc().
4.2	Designed a class that demonstrates the use of constructor and destructor.
4.3	Write a java program to demonstrate the implementation of abstract class.
5.	**Inheritance**
5.1	Write a java program to implement single level inheritance.
5.2	Write a java program to implement method overriding
5.3	Write a java program to implement multiple inheritance.
6.	**Packages and Arrays**
6.1	Create a package, Add the necessary classes and import the package in java class.
6.2	Write a java program to add two matrices and print the resultant matrix.
6.3	Write a java program for multiplying two matrices and print the product for the same.
7.	**Vectors and Multithreading**
7.1	Write a java program to implement the vectors.
7.2	Write a java program to implement thread life cycle.
7.3	Write a java program to implement multithreading.
8.	**File Handling**
8.1	Write a java program to open a file and display the contents in the console window.
8.2	Write a java program to copy the contents from one file to other file.
8.3	Write a java program to read the student data from user and store it in the file.
9.	**GUI and Exception Handling**
9.1	Design a AWT program to print the factorial for an input value.
9.2	Design an AWT program to perform various string operations like reverse string, string concatenation etc.
9.3	Write a java program to implement exception handling.
9.3	Write a java program to implement exception handling.

10.	**GUI Programming.**
10.1	Design an AWT application that contains the interface to add student information and display the same.
10.2	Design a calculator based on AWT application.
10.3	Design an AWT application to generate result marks sheet.

Semester – IV		
NAMEOF THE COURSE	**COMPUTER FORENSICS**	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC402	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1	To understand the procedures for identification, preservation, and extraction of electronic evidence, auditing and investigation of network and host system intrusions, analysis and documentation of information gathered.
CO 2	To prepare for all stages of an investigation – planning, detection, initial response and management interaction, investigate various media to collect evidence, report them in a way that would be acceptable in the court of law.
CO 3	Find vulnerabilities and security loopholes that facilitate attackers.

COURSE LEARNING OUTCOME:

CLO 1	Conduct digital investigations that conform to accepted professional standards and are based on the investigative process: identification, preservation, examination, analysis, and reporting;
CLO 2	Identify and document potential security breaches of computer data that suggest violations of legal, ethical, moral, policy, and/or societal standard.Apply a solid foundational grounding in computer networks, operating systems, filesystems, hardware, and mobile devices to digital investigations and to the protection of computer network resources from unauthorized activity
CLO 3	Access and critically evaluate relevant technical and legal information and emerging industry trends; and
CLO 4	Communicate effectively the results of a computer, network, and/or data forensic analysis verbally, in writing, and in presentations to both technical and lay audiences

Unit 1	**Introduction to Cyber Crimes, Computer Forensics and Investigations as a Profession (15 LECTURES)**
1.1	**Introduction to Cyber Crimes :** Internet, hacking, ethical hacking, need of ethical hacking, Black Hat vs. Gray Hat vs. White Hat, How is Ethical hacking different from security auditing and digital forensics? Virus, Obscenity, software piracy, Data encryption, decryption, compression.
1.2	**Computer Forensics and Investigations as a Profession:** Understanding Computer Forensics, Computer Forensics Versus Other Related Disciplines, A Brief History of Computer Forensics, Understanding Case Law, Developing Computer Forensics Resources, Preparing for computer investigation, Understanding Law Enforcement Agency Investigations, Following the Legal Processes, Understanding Corporate Investigations, Establishing Company Policies, Displaying Warning Banners, Designating an Authorized Requester, Conducting Security Investigations, Distinguishing Personal and Company Property.
Unit 2	**Understanding Forensic Investigations, Crime Scene Investigations (15 LECTURES)**
2.1	**Understanding Forensic Investigations:** Preparing a Computer Investigation, An Overview of a Computer Crime, An Overview of a Company Policy Violation, Taking a Systematic Approach, Assessing the Case, Planning Your Investigation, Securing Your Evidence.
2.2	**Crime Scene Investigations:** Employee Termination Cases, Internet Abuse Investigations, E-mail Abuse Investigations, Attorney-Client Privilege Investigations, Media Leak Investigations, Interviews and Interrogations in High-Tech Investigations, Conducting an Investigation, Gathering the Evidence, Understanding Bit-stream Copies, Acquiring an Image of Evidence Media, Using ProDiscover Basic to Acquire a USB Drive.
Unit 3	**The Investigator's Office and Laboratory, Data Acquisitions (15 LECTURES)**
3.1	**The Investigator's Office and Laboratory:** Understanding Forensics Lab Certification Requirements, Identifying Duties of the Lab Manager and Staff, Lab Budget Planning, Acquiring Certification and Training, Determining the Physical Requirements for a Computer Forensics Lab, Identifying Lab Security Needs, Conducting High-Risk Investigations, Using Evidence Containers, Overseeing Facility Maintenance, Considering Physical Security Needs, Auditing a Computer Forensics Lab, Using a Disaster Recovery Plan.
3.2	**Data Acquisitions**: Understanding Storage Formats for Digital Evidence, Raw Format, Proprietary Formats, Advanced Forensic Format, Determining the Best Acquisition

	Method, Contingency Planning for Image Acquisitions, Performing RAID Data Acquisitions, Remote Acquisition with ProDiscover.
UNIT 4	**Processing Crime and Incident Scenes, Computer Forensics Tools (15 LECTURES)**
4.1	**Processing Crime and Incident Scenes:** Identifying Digital Evidence, Understanding Rules of Evidence, Collecting Evidence in Private-Sector Incident Scenes, Processing Law Enforcement Crime Scenes, Understanding Concepts and Terms Used in Warrants, Preparing for a Search, Identifying the Nature of the Case, Identifying the Type of Computing System, Determining Whether You Can Seize a Computer, Obtaining a Detailed Description of the Location, Determining Who Is in Charge, Using Additional Technical Expertise, Determining the Tools You Need, Preparing the Investigation Team, Securing a Computer Incident or Crime Scene, Seizing Digital Evidence at the Scene, Preparing to Acquire Digital Evidence, Processing an Incident or Crime Scene, Processing Data Centers with RAID Systems, Using a Technical Advisor, Documenting Evidence in the Lab, Processing and Handling Digital Evidence, Storing Digital Evidence, Evidence Retention and Media Storage Needs, Documenting Evidence.
4.2	**Computer Forensics Tools:** Evaluating Computer Forensics Tool Needs, Types of Computer Forensics Tools, Tasks Performed by Computer Forensics Tools, Tool Comparisons, Computer Forensics Software Tools, Command-Line Forensics Tools, Other GUI Forensics Tools, Computer Forensics Hardware Tools, Forensic Workstations, Recommendations for a Forensic Workstation, Validating and Testing Forensics Software, Using National Institute of Standards and Technology (NIST) Tools, Using Validation Protocols.
UNIT 5	**Cell Phone and Mobile Device Forensics, Internet Forensics, Investigation, Evidence presentation and Legal aspects of Digital Forensics (15 LECTURES)**
5.1	**Cell Phone and Mobile Device Forensics**: Understanding Mobile Device Forensics, Mobile Phone Basics, Inside Mobile Devices, Inside PDAs, Acquisition Procedures for Cell Phones and Mobile Devices, Mobile Forensics Equipment.
5.2	**Internet Forensics**: E-mail Forensics: e-mail analysis, e-mail headers and spoofing, laws against e-mail Crime, Browser Forensics: Cookie Storage and Analysis, Analyzing Cache and temporary internet files, Web browsing activity reconstruction. **Investigation, Evidence**
5.3	**Presentation and Legal aspects of Digital Forensics:** Authorization to collect the evidence, acquisition of evidence, authentication of the evidence, analysis of the evidence, laws and regulations, Information

	Technology Act, Presenting evidence in court.

REFERENCES :

- Guide to Computer Forensics and Investigations, Bell Nelson, Amelia Phillips, Christopher Steuart, Fourth
- Computer Forensics: Computer Crime Scene Investigation, John R. Vacca, Second, Charles River Media
- Incident Response and computer forensics, Kevin Mandia, Chris Prosise, Second, Tata McGraw Hill

NAME OF THE COURSE	**COMPUTER FORENSICS**	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP402	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	3	
TOTAL NUMBER OF LECTURES PER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

PRACTICALS:

1.	File System Analysis using the Sleuth Kit.
2.	Using Data Acquisition tools.
3.	Using Forensic Toolkit (FTK).
4.	Using File Recovery tools.
5.	Forensic investigation using EnCase.
6.	Using Steganography tools.
7	Using Password cracking tools.
8	Using Log Capturing and Analysis tools.
9	Using Traffic Capturing and Analysis tools.

10	Using Wireless Forensics tools.
11	Using Web attack detection tools.
12	Using Email Forensic tools.
13	Using Mobile Forensic tools.
14	Capturing and analyzing network packets using Wireshark.
15	Analyze the packets provided in lab and solve the questions using Wireshark

Semester – IV		
NAMEOF THE COURSE	**ARTIFICIAL INTELLIGENCE**	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC403	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1	To present an overview of artificial intelligence (AI) principles and approaches with comprehensive and in-depth knowledge of AI principles and techniques by introducing conceptsAI's fundamental problems, and the state-of-the-art models and algorithms used to undertake these problems.
CO 2	Gain a historical perspective of AI and its foundations.
CO 3	Develop a basic understanding of the building blocks of AI as presented in terms of intelligent agents: Search, Knowledge representation, inference, logic, and learning.
CO 4	To present an overview of artificial intelligence (AI) principles and approaches with comprehensive and in-depth knowledge of AI principles and techniques by introducing conceptsAI's fundamental problems, and the state-of-the-art models and algorithms used to undertake these problems.
CO 5	Gain a historical perspective of AI and its foundations.

COURSE LEARNING OUTCOME:

CLO 1	Demonstrate fundamental understanding of the history of artificial intelligence (AI) and itsfoundations.
CLO 2	Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
CLO 3	To analyze the structures and algorithms of a selection of techniques related to searching,reasoning, machine learning, and language processing.
CLO 4	To define and analyze first order logic
CLO 5	To define planning algorithms and categorize knowledge representation

Unit 1	**Introduction, Intelligent Agents (15 LECTURES)**
1.1	**Introduction:** What is Artificial Intelligence? Foundations of AI, history, the state of art AI today.
1.2	**Intelligent Agents:** agents and environment, good behavior, nature of environment, the structure of agents.
Unit 2	**Solving Problems by Searching, Beyond Classical Search (15 LECTURES)**
2.1	**Solving Problems by Searching:** Problem solving agents, examples problems, searching for solutions, uninformed search, informed search strategies, heuristic functions.
2.2	**Beyond Classical Search:** local search algorithms, searching with non-deterministic action.
Unit 3	**Adversarial Search, Logical Agents (15 LECTURES)**
3.1	**Adversarial Search:** Games, optimal decisions in games, stochastic games, partially observable games, state-of-the-are game programs.
3.2	**Logical Agents:** Knowledge base agents, The Wumpus world, logic, propositional logic, propositional theorem proving, effective propositional model checking.
Unit 4	**First Order Logic, Inference in First Order Logic (15 LECTURES)**
4.1	**First Order Logic:** Syntax and semantics, using First Order Logic, Knowledge engineering in First Order Logic.
4.2	**Inference in First Order Logic:** propositional vs. First Order, unification and lifting, forward and backward chaining, resolution.
Unit 5	**Planning, Knowledge Representation (15 LECTURES)**
5.1	**Planning:** Definition of Classical Planning, Algorithms for planning as state space search, planning graphs, analysis of planning approaches
5.2	**Knowledge Representation:** Categories and Objects, events, reasoning systems for categories, Internet shopping world

REFERENCES:

- Artificial Intelligence: A Modern Approach Stuart Russel and Peter Norvig Pearson 3rd 2015
- A First Course in Artificial Intelligence Deepak Khemani TMH First 2017
- Artificial Intelligence: A Rational Approach Rahul Deva Shroff publishers 1st 2018

NAME OF THE COURSE	**ARTIFICIAL INTELLIGENCE**	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP403	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	3	
TOTAL NUMBER OF LECTURES PER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

PRACTICALS

1.1	Write a program to implement depth first search algorithm.
1.2	Write a program to implement breadth first search algorithm.
2.1	Write a program to simulate 4-Queen / N-Queen problem.
2.2	Write a program to solve tower of Hanoi problem.
3.1	Write a program to implement alpha beta search.
3.2	Write a program for Hill climbing problem.
4.1	Write a program to implement A* algorithm.
4.2	Write a program to implement AO* algorithm.
5.1	Write a program to solve water jug problem.
5.2	Design the simulation of tic – tac – toe game using min-max algorithm.
6.1	Write a program to solve Missionaries and Cannibals problem.
6.2	Design an application to simulate number puzzle problem.
7.1	Write a program to shuffle Deck of cards.
7.2	Solve traveling salesman problem using artificial intelligence technique.
8.1	Solve the block of World problem.
8.2	Solve constraint satisfaction problem
9.1	Derive the expressions based on Associative law
9.2	Derive the expressions based on Distributive law
10.1	Write a program to derive the predicate. (for e.g.: Sachin is batsman , batsman is cricketer) - > Sachin is Cricketer.
10.2	Write a program which contains three predicates: male, female, parent. Make rules for following family relations: father, mother, grandfather, grandmother, brother, sister, uncle, aunt, nephew and niece, cousin. Question: Draw Family Tree. Define: Clauses, Facts, Predicates and Rules with conjunction and disjunction

Semester – IV		
NAMEOF THE COURSE	**IT SERVICE MANAGEMENT**	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC404	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1	To be able to identify and illustrate basic terminology and concepts ofITSM
CO 2	To be able to describe the functions, roles and processes for each of the phases of theITIL Service Lifecycle.
CO 3	Apply a service-oriented approach to business systems design and operations in orderthat an organization is more efficient and effective.
CO 4	State the activities under taken in service operation/explain, analyze, and critique theconcept of IT Service Management taking an example.
CO 5	To identify the importance of process improvement and would be able to state the variousstеps in it.

COURSE LEARNING OUTCOMES:

CLO 1	Describe the importance of service management and associated 4p's giving example.
CLO 2	describe using suitable example the ITIL service lifecycle
CLO 3	state the activities undertaken in service design of an application.
CLO 4	state the process of incident reporting
CLO 5	taking a suitable example explain RACI model

Unit 1	**IT Service Management, Service Strategy Principles, Service Strategy (15 LECTURES)**
1.1 1.2 1.3	**IT Service Management:** Introduction, What is service management? What are services? Business Process, Principles of Service management: Specialization and Coordination, The agency principle, Encapsulation, Principles of systems, The service Life Cycle, Functions and processes across the life cycle. **Service Strategy Principles:** Value creation, Service Assets, Service Provider Service Structures, and Service Strategy Principles. **Service Strategy:** Define the market, Develop the offerings, Develop Strategic Assets, Prepare for execution.
Unit 2	**Service Design, Service Design Principles, Service Design Processes (15 LECTURES)**
2.1 2.2	**Service Design:** Fundamentals, **Service Design Principles:** Goals, Balanced Design, Identifying Service requirements, identifying and documenting business requirements and drivers, Design activities, Design aspects, Subsequent design activities, Design constraints, Service oriented architecture, Business Service Management, Service Design Models. **Service Design Processes:** Service Catalogue Management, Service Level Management, Capacity Management, Availability Management, IT Service Continuity Management, Information Security Management, Supplier Management
Unit 3	**Service Transition, Service Transition Principles, Service Transition Processes (15 LECTURES)**
3.1 3.2	**Service Transition:** Fundamentals, **Service Transition Principles:** Principles Supporting Service Transition, Policies for Service Transition **Service Transition Processes:** Transition planning and support, Change Management, Service Asses Configuration Management, Service and Deployment Management, Service Validation and Testing, Evaluation, Knowledge Management.
Unit 4	**Service Operation, Service Operation Principles, Service Operation Processes (15 LECTURES)**
4.1	**Service Operation:** Fundamentals, **Service Operation Principles:** Functions, groups, teams, departments and divisions, Achieving balance in service operations, Providing service, Operation staff involvement in service design and service transition, Operational Health, Communication, Documentation

4.2	**Service Operation Processes:** Event Management, Incident Management, Request fulfilment, Problem Management, Access Management, Operational activities of processes covered in other lifecycle phases.
Unit 5	**Continual Service Improvement (CSI) Principles, CSI Methods and Techniques, Organising for CSI, Implementing CSI (15 LECTURES)**
5.1 5.2 5.3	**Continual Service Improvement(CSI) Principles:** CSI Approach, CSI and organizational change, Ownership, CSI register, External and Internal drivers, Service level management, Knowledge management, The Deming cycle, Service Measurement, IT governance, Frameworks, models, standards and quality Systems, CSI inputs and outputs. CSI Process: The seven step improvement process. **CSI Methods and Techniques:** Methods and techniques, Assessments, benchmarking, Service Measurement, Metrics, Return on Investment, Service reporting, CSI and other service management processes, **Organizing for CSI:** Organizational development, Functions, roles, Customer Engagement, Responsibility model - RACI, Competence and training, Implementing CSI: Critical Considerations for implementing Scythe start, Governance, CSI and organizational change, Communication Strategy and Plan

REFERENCES:

- ITIL v3 Foundation Complete Certification Kit
- Michael Burton, Gerhard Franken, Android application development for dummies, 2nd ed, John Wiley & Sons
- Ted Hagos, Android Studio IDE Quick Reference: A Pocket Guide to Android Studio Development, Apress

NAME OF THE COURSE	**ADVANCED MOBILE PROGRAMMING PRACTICAL**	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP404	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	3	
TOTAL NUMBER OF LECTURES PER SEMESTER	45	
EVALUATION METHOD	INTERNAL	SEMESTER END
	ASSESSMENT	EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

Practical No	**Details**
1	Introduction to Android, Introduction to Android Studio IDE, Application Fundamentals: Creating a Project, Android Components, Activities, Services, Content Providers, Broadcast Receivers, Interface overview, Creating Android Virtual device, USB debugging mode, Android Application Overview. Simple "Hello World" program.
2	Programming Resources Android Resources: (Color, Theme, String, Drawable, Dimension, Image),
3	Programming Activities and fragments Activity Life Cycle, Activity methods, Multiple Activities, Life Cycle of fragments and multiple fragments.
4	Programs related to different Layouts Coordinate, Linear, Relative, Table, Absolute, Frame, List View, Grid View.
5	Programming UI elements AppBar, Fragments, UI Components
6	Programming menus, dialog, dialog fragments
7	Programs on Intents, Events, Listeners and Adapters The Android Intent Class, Using Events and Event Listeners
8	Programs on Services, notification and broadcast receivers
9	Database Programming with SQLite
10	Programming threads, handles and asynchronized programs
11	Programming Media API and Telephone API
12	Programming Security and permissions

Semester – IV		
NAMEOF THE COURSE	**COMPUTER GRAPHICS AND ANIMATION**	
CLASS	SYBSc IT	
COURSE CODE	SBTTEC405	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	5	
TOTAL NUMBR OF LECTURES PER SEMESTER	75	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESER END EXAMINATION
TOTAL MARKS	50	50
PASSING MARKS	20	20

COURSE OBJECTIVES:

CO 1.	To learn the fundamentals of computer graphics and scan conversion algorithms.
CO 2.	To learn Geometrical Transformations in 2-Dimensional and 3-Dimensional perspectives.
CO 3.	To learn stages in 3D viewing, Canonical View Volume, Radiometry, Colorimetry, Color Spaces, Color Appearance
CO 4.	To learn visible-surface determination algorithms, Curve Representation, Bezier Curves, B-spline Curves.
CO 5.	To learn Principles of Animation, Key framing, Image, Digital image file formats, Image compression standard

COURSE LEARNING OUTCOMES:

CLO 1.	Explore the structure of an interactive computer graphics system, and the separation of system components.
CLO 2.	Apply the concept of 2D and 3D geometrical transformations
CLO 3.	Implement the knowledge of viewing in 3D, Canonical View Volume,

	Radiometry, Photometry.
CLO 4.	Get familiar with Visible-Surface Determination algorithm and Curve Representation.
CLO 5.	Get accustomed to Principles of Animation, Image Manipulation and Storage.

Unit 1	**Introduction to Computer Graphics,Scan conversion (15 LECTURES)**
1.1	**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.
1.2	**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.
Unit 2	**Two-Dimensional Transformationsand Three-Dimensional Transformations (15 LECTURES)**
2.1	**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.

2.2	**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.
Unit 3	**Viewing in 3D, Light, color (15 LECTURES)**
3.1	**Viewing in 3D:** Stages in 3D viewing, Canonical View Volume (CVV), Specifying an Arbitrary 3D View, Examples of 3D Viewing, The Mathematics of Planar Geometric Projections, Combined transformation matrices for projections and viewing, Coordinate Systems and matrices, camera model and viewing pyramid.
3.2	**Light:** Radiometry, Transport, Equation, Photometry
3.3	**Color:** Colorimetry, Color Spaces, Chromatic Adaptation, Color Appearance
Unit 4	**Visible-Surface Determination, Plane Curves and Surfaces (15 LECTURES)**
4.1	**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.
4.2	**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.

Unit 5	Computer Animation and Image Manipulation and Storage (15 LECTURES)
5.1	**Computer Animation:** Principles of Animation, Key framing, Deformations, Character Animation, Physics- Based Animation, Procedural Techniques, Groups of Objects.
5.2	**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.

REFERENCES:

- Computer graphics. 2nd ed. Mishra, Ruchi, Global Academic Publishers & Distributors 2015
- Computer graphics. Mishra, Ruchi Wiley India, 2011
- Computer graphics with virtual reality systems, Maurya, Rajesh K. Wiley India 2009
- Fundamentals of computer graphics. 4th ed. Marschner, Steve & Shirley, Peter CRC Press / Taylor and Francis Group 2016

NAME OF THE COURSE	**Computer Graphics and Animation**	
CLASS	SYBSCIT	
COURSE CODE	SBTTECP405	
NUMBER OF CREDITS	2	
NUMBER OF LECTURES PER WEEK	3	
TOTAL NUMBER OF LECTURES PER SEMESTER	45	
EVALUATION METHOD	INTERNAL ASSESSMENT	SEMESTER END EXAMINATION
TOTAL MARKS	---	50
PASSING MARKS	---	20

List of Practical	
1.	**Solve the following:**
1.1	Study and enlist the basic functions used for graphics in C / C++ / Python language. Give an example for each of them.
1.2	Draw a co-ordinate axis at the center of the screen.

2.	**Solve the following:**
2.1	Divide your screen into four region, draw circle, rectangle, ellipse and half ellipse in each region with appropriate message.
2.2	Draw a simple hut on the screen.
3.1	**Draw the following basic shapes in the center of the screen :**
4.	**Solve the following:**
4.1	Develop the program for DDA Line drawing algorithm.
4.2	Develop the program for Bresenham's Line drawing algorithm.
5	**Solve the following:**
5.1	Develop the program for the mid-point circle drawing algorithm.
5.2	Develop the program for the mid-point ellipse drawing algorithm.
6	**Solve the following:**
6.1	Write a program to implement 2D scaling.
6.2	Write a program to perform 2D translation
7	**Solve the following:**
7.1	Perform 2D Rotation on a given object.
7.2	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:**
8.1	Write a program to implement Cohen-Sutherland clipping.
8.2	Write a program to implement Liang - Barsky Line Clipping Algorithm
9	**Solve the following:**
9.1	Write a program to fill a circle using Flood Fill Algorithm.
9.2	Write a program to fill a circle using Boundary Fill Algorithm.
10	**Solve the following:**
10.1	Develop a simple text screen saver using graphics functions.
10.2	Perform smiling face animation using graphic functions.
10.3	Draw the moving car on the screen.

ASSESSMENT DETAILS:(this will be same for all the theory papers)

Internal Assessment (50 marks)

Part 1: Project Work/Assignment (**25 Marks**) & Test (**25 Marks**)

- At the beginning of the semester, students should be assigned project topics drawn from Unit 1 to Unit 5.
- Students can work in groups of not more than 3 per topic.
- Project Marks will be divided as written submission: 10 Marks & Presentation & Viva: 15 marks)
- The Project/Assignment can take the form of Street-Plays/Power-Point Presentations/Poster Exhibitions and similar other modes of presentation appropriate to the topic.
- Students must submit the Project/Assignment before the last teaching day of the semester.

Semester End Examination – External Assessment (50 marks)

- The duration of the paper will be two hours.
- There shall be five compulsory questions
- Q1-5 shall correspond to the five units. Q1-5 shall contain an internal choice (attempt any 2 of 4). Q1-5 shall carry a maximum of 10 marks

Practical Assessment (for papers with practicals)

- The duration of the practical exam will be two and a half hours.
- The students are allowed to write the paper if the attendance for practicals is more than 75%
- To appear in the practical exam, students must bring a properly certified journal.

Evaluation Scheme:

1. **Internal Evaluation (25 Marks).**
   - **i. Test: 1 Class test of 25 marks. (Can be taken online)**

Q	**Attempt *any five* of the following:**	**25**
a.		
b.		
c.		
d.		
e.		
f.		
g.		

2. **External Examination: (50 marks)**

	All questions are compulsory	

Q1	**(Based on Unit 1) Attempt *any two* of the following:**	**10**
a.		
b.		
c.		
d.		
Q2	**(Based on Unit 2) Attempt *any two* of the following:**	**10**
Q3	**(Based on Unit 3) Attempt *any two* of the following:**	**10**
Q4	**(Based on Unit 4) Attempt *any two* of the following:**	**10**
Q5	**(Based on Unit 5) Attempt *any two* of the following:**	**10**

3. Practical Exam: 50 marks

A Certified copy journal is essential to appear for the practical examination.

1.	**Practical Question 1**	**20**
2.	**Practical Question 2**	**20**
3.	**Journal**	**5**
4.	**Viva Voce**	**5**

OR

1.	**Practical Question**	**40**
2.	**Journal**	**5**
3.	**Viva Voce**	**5**