SIES COLLEGE OF COMMERCE & ECONOMICS
AUTONOMOUS
DEPARTMENT OF INFORMATION TECHNOLOGY

Sr. No.	Heading	Particulars
1	Title of the course	B. Sc. (Information Technology)
2	Eligibility for admission	HSC or Equivalent with Mathematics as Compulsory Subject
3	Minimum percentage	45 %
4	Semesters	III & IV
5	Level	UG
6	Pattern	03 years & 06 semesters CBGS
7	To be implemented from	From Academic year 2021-22 in a progressive manner

(WITH EFFECT FROM THE ACADEMIC YEAR 2021-2022)

Semester – 3			
Course Code	**Course Type**	**Course Title**	**Credits**
BITS301	Skill Enhancement Course	Python Programming	2
BITS302	Core Subject	Data Structures	2
BITS303	Core Subject	Computer Networks	2
BITS304	Core Subject	Database Management Systems	2
BITS305	Core Subject	Computer Oriented Statistical Techniques	2
BITS3P1	Skill Enhancement Course Practical	Python Programming Practical	2
BITS3P2	Core Subject Practical	Data Structures Practical	2
BITS3P3	Core Subject Practical	Computer Networks Practical	2
USIT3P4	Core Subject Practical	Database Management Systems Practical	2
BITS3P5	Core Subject Practical	Mobile Programming Practical	2
		Total Credits	**20**

Semester – 4			
Course Code	**Course Type**	**Course Title**	**Credits**
BITS401	Skill Enhancement Course	Core Java	2
BITS402	Core Subject	Introduction to Embedded Systems	2
BITS403	Core Subject	Applied Mathematics	2
BITS404	Core Subject	Software Engineering	2
BITS405	Core Subject	Computer Graphics and Animation	2
BITS4P1	Skill Enhancement Course Practical	Core Java Practical	2
BITS4P2	Core Subject Practical	Introduction to Embedded Systems Practical	2
BITS4P3	Core Subject Practical	Computer Oriented Statistical Techniques Practical	2
BITS4P4	Core Subject Practical	Software Engineering Practical	2
BITS4P5	Core Subject Practical	Computer Graphics and Animation Practical	2
		Total Credits	**20**

SEMESTER III

Python Programming

COURSE CODE: BITS301 **COURSE CREDIT: 02**

Course Objectives:

- To explain a basic introduction to object-oriented and procedural programming using Python.
- To acquire knowledge and programming skills in python to solve problems in different domains.

Sr. No	Modules/Units	No of Lectures
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, **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
2.	**Functions:** Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution, Parameters and Arguments, Variables and Parameters Are Local, Stack Diagrams, Fruitful Functions and Void Functions, Why Functions? Importing with from, Return Values, Incremental Development, Composition, Boolean Functions, More Recursion, Leap of Faith, Checking Types **Strings**: A String Is a Sequence, Traversal with a for Loop, String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, The in Operator, String Comparison, String Operations.	12
3.	**Lists:** Values and Accessing Elements, Lists are mutable, traversing a List, Deleting elements from List, Built-in List Operators, Concatenation, Repetition, In Operator, Built-in List functions and methods **Tuples and Dictionaries**: Tuples, Accessing values in Tuples, Tuple Assignment, Tuples as return values, Variable-length argument tuples, Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in Tuple Functions Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods **Files:** Text Files, The File Object Attributes, Directories **Exceptions:** Built-in Exceptions, Handling Exceptions, Exception,	12

	withArguments, User-defined Exceptions	
4.	**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
5.	**Creating the GUI Form and Adding Widgets:** **Widgets**: Button, Canvas, Checkbutton, Entry, Frame, Label, Listbox Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel, Spinbox, PanedWindow, LabelFrame, tkMessagebox. Handling Standard attributes and Properties of Widgets. **Layout Management**: Designing GUI applications with proper Layout Management features. **Look and Feel Customization**: Enhancing Look and Feel of GUI using different appearances of widgets. **Storing Data in Our MySQL Database via Our GUI :** Connecting to a MySQL database from Python, Configuring the MySQL connection, Designing the Python GUI database, Using the INSERT command, Using the UPDATE command, Using the DELETE command, Storing and retrieving data from MySQL database.	12

REFERENCE BOOKS:

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	Jason Montojo, Jennifer Campbell, and Paul Gries	SPD	1	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

Python Programming Practical

COURSE CODE: BITS3P1 **COURSE CREDIT: 02**

Course Objectives:

- To demonstrate the principles of object oriented programming in well-written modular code.
- To enable students to solve problems requiring the writing of well-documented programs in the Python language.

List of Practical	
1.	**Write the program for the following:**
a.	Create a program that asks the user to enter their name and their age. Print out amessage 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 orodd, 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 Trueif 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 thefollowing: **** ********* *******
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 onecommon member.
b.	Write a Python program to print a specified list after removing the 0th, 2nd, 4thand 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 xand y(these shouldall be numbers). i. Write a method called addwhich returns the sum of the attributes xand y. ii. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. iii. Write a static method called subtract, which takes two number parameters, band c, and returns b - c. iv. Write a method called value which returns a tuple containing the values of xand y. Make this method into a property, and write a setter and a deleter for manipulating the values of xand y.
8.	**Write the program for the following:**
a.	Open a new file in IDLE (“New Window” in the “File” menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the “Control Flow and Functions” exercise into this file and save it. Now open a new file and save it in the same directory. You should now be able

	to import your own module like this: import geometry Try and add print dir(geometry)to the file and run it. Now write a function pointyShapeVolume(x, y, squareBase) that calculates the volume of a square pyramid if squareBase is True and of a right circular cone if squareBase is False. x is the length of an edge on a square if squareBase is True and the radius of a circle when squareBase is False. y is the height of the object. First use squareBase to distinguish the cases. Use the circleArea and squareArea from the geometry module to calculate the base areas.
b.	Write a program to implement exception handling.
9.	**Write the program for the following:**
a.	Try to configure the widget with various options like: bg=”red”, family=”times”,size=18
b.	Try to change the widget type and configuration options to experiment with otherwidget 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 thesame.
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 modifytherecords.

REFERENCE BOOKS:

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

Data Structures

COURSE CODE: BITS302 **COURSE CREDIT: 02**

Course Objectives:

- To enable students to understand the representation and use of primitive data types, built in data structures and allocation used in memory.
- To enable students to understand the concept of stack, queue, link list, tree, graph, memory allocation, garbage collection and applications of Data Structures.

Sr. No	Modules/Units	No of Lectures
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. **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, Sparse Matrix, Memory Representation of Special kind of Matrices, Advantages and Limitations of Arrays.	12
2.	**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.	12
3.	**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

4.	**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
5.	**Hashing Techniques** Hash function, Address calculation techniques, Common hashing functions Collision resolution, Linear probing, Quadratic, Double hashing, Bucket hashing, 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, Graph Traversal, Applications of the Graph, Reachability, Shortest Path Problems, Spanning Trees.	12

REFERENCE BOOKS:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A Simplified Approach to Data Structures	Lalit Goyal, Vishal Goyal, Pawan Kumar	SPD	1st	2014
2.	An Introduction to Data Structure with Applications	Jean – Paul Tremblay and Paul Sorenson	Tata MacGraw Hill	2nd	2007
3.	Data Structure and Algorithm	Maria Rukadikar	SPD	1st	2017
4.	Schaum‟s Outlines Data structure	Seymour Lipschutz	Tata McGraw Hill	2nd	2005
5.	Data structure – A Pseudocode Approach with C	AM Tanenbaum, Y Langsam and MJ Augustein	Prentice Hall India	2nd	2006
6.	Data structure and Algorithm Analysis in C	Weiss, Mark Allen	Addison Wesley	1st	2006

Data Structures Practical

COURSE CODE: BITS3P2 **COURSE CREDIT: 02**

Course Objectives:

- To help students to learn programming various inserting, deleting, sorting, searching, traversing mechanisms with various data structures.

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

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

REFERENCE BOOKS:

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

Computer Networks

COURSE CODE: BITS303 **COURSE CREDIT: 02**

Course Objectives:

- To help students acquire basic knowledge about data communications and computer networking.
- To assist student to learn about the different models and devices related to networks.

Sr. No	Modules/Units	No of Lectures
1.	**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
2.	**Bandwidth Utilization: Multiplexing and Spectrum Spreading:** Multiplexing, Spread Spectrum **Transmission media:** Guided Media**,** Unguided Media **Switching:** Introduction, circuit switched networks, packet switching, structure of a switch. **Introduction to the Data Link Layer:** Link layer addressing, Data Link Layer Design Issues, Error detection and correction, block coding, cyclic codes, checksum, forward error correction, error correcting codes, error detecting codes.	12
3.	**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
4.	**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

5.	**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, **Standard Client0Server Protocols:** World wide-web and HTTP, FTP, Electronic mail, Telnet, Secured Shell, Domain name system.	12

REFERENCE BOOKS:

Sr. No.	**Title**	**Authors**	**Publisher**
1.	Data Communication & Networking	Behrouz A. Forouzan	Tata McGraw Hill
2.	TCP/IP Protocol Suite	Behrouz A. Forouzan	Tata McGraw Hill
3.	Computer Networks	Andrew Tanenbaum	Pearson

Computer Networks Practical

COURSE CODE: BITS3P3 **COURSE CREDIT: 02**

Course Objectives:

- To enable students to simulate the working of a network topology.
- To enable students to analyze packets in a network.

List of Practical	
1.	IPv4 Addressing and Subnetting.
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

Database Management Systems

COURSE CODE: BITS304 **COURSE CREDIT: 02**

Course Objectives:

- To help students to learn database management system with an emphasis on how to organize, maintain and retrieve information from a DBMS.
- To help students to learn about ER Diagram and their relationships.
- To help students learn the concepts of integrity and security.

Sr. No	Modules/Units	No of Lectures
1.	**Introduction to Databases and Transactions :** What is database system, purpose of database system, view of data, relational databases, database architecture, transaction management **Data Models:** The importance of data models, Basic building blocks, Business rules, The evolution of data models, Degrees of data abstraction. **Database Design, ER Diagram and Unified Modelling 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	12
2.	**Relational database model:** Logical view of data, keys, integrity rules, Relational Database design: features of good relational database design, atomic domain and Normalization (1NF, 2NF, 3NF, BCNF). **Relational Algebra and Calculus** **Relational algebra:** introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison. **Relational Calculus:** Tuple relational calculus, Domain relational Calculus, calculus vs algebra, computational capabilities	12

	Constraints, Views and SQL Constraints, types of constrains, Integrity constraints, Views: Introduction to views, data independence, security, updates on views, comparison between tables and views SQL: data definition, aggregate function, Null Values, nested sub queries, Joined relations. Triggers.	12
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.	12
5.	**PL-SQL:** Beginning with PL / SQL, Identifiers and Keywords, Operators, Expressions, Sequences, Control Structures, Cursors and Transaction, Collections and composite data types, Procedures and Functions, Exceptions Handling, Packages, With Clause and Hierarchical Retrieval.	12

REFERENCE BOOKS:

Sr. No.	**Title**	**Authors**	**Publisher**	**Edition**	**Year**
1.	Database System and Concepts	A Silberschatz, H Korth, S Sudarshan	McGrawHill	Fifth Edition	-
2.	Database Systems	Rob Coronel	Cengage Learning	Twelfth Edition	-
3.	Programming with PL/SQL for Beginners	H. Dand, R. Patil and T. Sambare	X –Team	First	2011
4.	Introduction to Database System	C.J.Date	Pearson	First	2003

Database Management Systems Practical

COURSE CODE: BITS3P4COURSE CREDIT: 02

Course Objectives:

- To make students learn basic SQL queries to retrieve, delete, update and insert the data in database.
- To make students learn to develop skills for query processing and optimization.
- To make students learn to identify the basic issues of transaction processing.

List of Practical	
1.	**SQL Statements – 1**
a.	Writing Basic SQL SELECT Statements
b.	Restricting and Sorting Data
c.	Single-Row Functions
2.	**SQL Statements – 2**
a.	Displaying Data from Multiple Tables
b.	Aggregating Data Using Group Functions
c.	Subqueries
3.	**Manipulating Data**
a.	Using INSERT statement
b.	Using DELETE statement
c.	Using UPDATE statement
4.	**Creating and Managing Tables**
a.	Creating and Managing Tables
b.	Including Constraints
5.	**Creating and Managing other database objects**
a.	Creating Views
b.	Other Database Objects
c.	Controlling User Access
6.	**Using SET operators, Date/Time Functions, GROUP BY clause (advanced features) and advanced subqueries**
a.	Using SET Operators
b.	Datetime Functions

c.	Enhancements to the GROUP BY Clause
d.	Advanced Subqueries
7.	**PL/SQL Basics**
a.	Declaring Variables
b.	Writing Executable Statements
c.	Interacting with the Oracle Server
d.	Writing Control Structures
8.	**Composite data types, cursors and exceptions.**
a.	Working with Composite Data Types
b.	Writing Explicit Cursors
c.	Handling Exceptions
9.	**Procedures and Functions**
a.	Creating Procedures
b.	Creating Functions
c.	Managing Subprograms
d.	Creating Packages
10.	**Creating Database Triggers**

REFERENCE BOOKS:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Database System and Concepts	A Silberschatz,H Korth, S Sudarshan	McGraw-Hill	Fifth	-
2.	Programming with PL/SQL for Beginners	H.Dand , R.Patil and T. Sambare	X –Team	First	2011
3.	PL/SQL Programming	Ivan Bayross	BPB	First	2010

Computer Oriented Statistical Techniques

COURSE CODE: BITS305 **COURSE CREDIT: 02**

Course Objectives:

- Students will be taught to analyze ungrouped and grouped data using measures of location anddispersion.
- Students will be taught to perform test of hypothesis as well as calculate confidence interval for apopulation parameter for single sample and double sample.
- Students will be taught to apply t-test and Chi-Square test for independence and Goodness of fit.
- Students will be taught to compute and interpret results of bivariate and multivariate regression andcorrelation analysis for forecasting.

Sr. No	Modules/Units	No of Lectures
1.	**Presentation of Data :** Univariate Frequency Distribution of Discreteand Continuous Variables, Cumulative Frequency Distribution, Graphical representation of Frequency Distribution by Histogram, Frequency Polygon, Frequency Curve And Ogives. Diagrammatic representation using Bar Diagrams and Pie Chart. **Measures of Central Tendency**: Introduction, Arithmetic Mean and itsProperties (Simple and Weighted), Geometric mean and Harmonic mean, Quantiles (Median, Quartiles, Deciles, and Percentiles), Mode, Empirical Relation Between Mean, Median, and Mode, Relation Between the Arithmetic, Geometric, and Harmonic Mean,The Root Mean Square. Merits, Demerits and Uses of Mean, Median, Mode, G.M. and H.M **Introduction to R:** Basic syntax, data types, variables, operators, controlstatements, R-functions, R –Vectors, R – lists, R Arrays.	12
2.	**Measures of Dispersion**: Introduction, Absolute and Relative Measures: Range, Interquartile Range, 10–90 Percentile Range , Quartile Deviation Mean Absolute Deviation, Standard Deviation, Variance and their relativemeasures. Empirical Relations Between Measures of Dispersion. **Moments, Skewness, and Kurtosis**: Moments, Relations BetweenMoments, Skewness, Kurtosis.	12
3.	**Elementary Sampling Theory** : Sampling Theory, Random Samples andRandom Numbers, Sampling With and Without Replacement, Sampling Distributions, Sampling Distribution of Means, Sampling Distribution of Proportions, Sampling Distributions of Differences andSums, Standard Errors. **Statistical Estimation Theory**: Estimation of Parameters, Unbiased Estimates, Efficient Estimates, Point Estimates and Interval Estimates, Confidence-Interval Estimates of Population Parameters, Probable Error.**Statistics in R:** mean, median, mode, Normal Distribution , Binomial Distribution, Frequency Distribution in R.	12

4.	**Introduction to Statistical Hypothesis Testing:** Statistical Hypotheses, Tests of Hypotheses and Significance, or Decision Rules, Type I and TypeII Errors, Level of Significance, Critical Region, Two-Tailed and One- Tailed Tests, The Power of a Test, p-Values for Hypotheses Tests. **Tests Involving Normal Distributions:** Test for Single Mean, SingleProportion, Test of Significance for Differences of Means, Test of Significance for Differences of Proportion. **Small Sample Tests**: 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, Definitionof Chi-Square, The Chi-Square Test for Goodness of Fit and Independence of Attributes, Contingency Tables, Yates' Correction, Coefficient of Contingency, Correlation of Attributes, Additive Property of Chi-Square.	12
5.	**Linear Correlation:** Introduction, Types of Correlation, Determination ofCorrelation using Scatter Diagram, Karl Pearson's Coefficient of Correlation and Spearman's Rank Correlation Coefficient. **Linear Regression:** Introduction, Regression Lines and Regression Coefficients, Relation between Coefficient of Correlation and RegressionCoefficients. **Curve Fitting and the Method of Least Squares:** Introduction, Freehand Method of Curve Fitting, The Straight Line Fitting by The Method of Least Squares, Nonlinear Relationships, Fitting of Second Degree Curve by The Method of Least-Squares. Applications to TimeSeries. **Sampling Theory of Correlation and Regression**	12

REFERENCE BOOKS:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Theory and Problems of Statistics	Schaum's Outlines Series, Murray R. Spiegel, Larry J. Stephens	McGraw-Hill	Sixth	2008
2.	Fundamental of Mathematical Statistics	S.C. Gupta and V.K. Kapoor	Sultan Chand and Sons	First	2011
3.	Introduction to Mathematical Statistics	Robert V. Hogg	Allen T. Craig	First	2010
4.	A Practical Approach using R	R.B. Patil, H.J. Dand and R. Bhavsar	Shroff Publishers and Distributors	First	2017

Computer Oriented Statistical Techniques Practical

COURSE CODE: BITS3P5 **COURSE CREDIT: 02**

Course Objectives:

- Students will be taught the basic syntax of R programming.
- Students will be taught to analyze data using statistical functions in R.
- Students will be taught to import, review, manipulate and summarize data-sets in R.
- Students will be taught to perform appropriate statistical tests using R.

List of Practical:	
1.	Using R execute the basic commands, array, list and frames.
2.	Create a Matrix using R and Perform the operations addition, inverse, transpose and multiplication operations.
3.	Using R Execute the statistical functions: mean, median, mode, quartiles, range, inter quartile range histogram
4.	Using R import the data from Excel / .CSV file and Perform the above functions.
5.	Using R import the data from Excel / .CSV file and Calculate the standard deviation, variance, co-variance.
6.	Using R import the data from Excel / .CSV file and draw the skewness.
7.	Using R perform the binomial and normal distribution on the data.
8.	Import the data from Excel / .CSV and perform the hypothetical testing.
9.	Import the data from Excel / .CSV and perform the Chi-squared Test.
10.	Perform the Linear Regression using R.

SEMESTER IV

Core Java

COURSE CODE: BITS401 **COURSE CREDIT: 02**

Course Objectives:

- To provide knowledge about basic Java language syntax and semantics to write Java programs.
- To assist students to understand the fundamentals of object-oriented programming in Java to design GUI applications
- To teach how to design a graphical user interface (GUI) using applets and AWT in Java.

Sr. No	Modules/Units	No of Lectures
1.	**Introduction**: History, architecture and its components, Java Class File, Java Runtime Environment, The Java Virtual Machine, JVM Components, The Java API, java platform, java development kit, Lambda Expressions, Methods References, Type Annotations, Method Parameter Reflection, setting the path environment variable, Java Compiler And Interpreter, java programs, java applications, main(), public, static, void, string[] args, statements, white space, case sensitivity, identifiers, keywords, comments, braces and code blocks, variables, variable name **Data types**: primitive data types, Object Reference Types, Strings, Auto boxing, operators and properties of operators, Arithmetic operators, assignment operators, increment and decrement operator, relational operator, logical operator, bitwise operator, conditional operator.	12
2.	**Control Flow Statements**: The If…Else If…Else Statement, The Switch…Case Statement **Iterations**: The While Loop, The Do … While Loop, The For Loop, The Foreach Loop, Labeled Statements, The Break And Continue Statements, The Return Statement **Classes**: Types of Classes, Scope Rules, Access Modifier, Instantiating Objects From A Class, Initializing The Class Object And Its Attributes, Class Methods, Accessing A Method, Method Returning A Value, Method's Arguments, Method Overloading, Variable Arguments [Varargs], Constructors, this Instance, super Instance, Characteristics Of Members Of A Class, constants, this instance, static fields of a class, static methods of a class, garbage collection.	12
3.	**Inheritance**: Derived Class Objects, Inheritance and Access Control, Default Base Class Constructors, this and super keywords. Abstract Classes And Interfaces, Abstract Classes, Abstract Methods, Interfaces, What Is An Interface? How Is An Interface Different From An Abstract Class?, Multiple Inheritance, Default Implementation, Adding New Functionality, Method Implementation, Classes V/sInterfaces, Defining An Interface, Implementing Interfaces.	12

	Packages: Creating Packages, Default Package, Importing Packages, Using A Package.	
4.	**Enumerations, Arrays**: Two Dimensional Arrays, Multi-Dimensional Arrays, Vectors, Adding Elements To A Vector, Accessing Vector Elements, Searching For Elements In A Vector, Working With The Size of The Vector. **Multithreading**: the thread control methods, thread life cycle, the main thread, creating a thread, extending the thread class. **Exceptions**: Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause **Byte streams**: reading console input, writing console output, reading file, writing file, writing binary data, reading binary data, getting started with character streams, writing file, reading file	12
5.	**Event Handling:** Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes and inner classes. **Abstract Window Toolkit:** Window Fundamentals, Component, Container, Panel, Window, Frame, Canvas.Components – Labels, Buttons, Check Boxes, Radio Buttons, Choice Menus, Text Fields, Text, Scrolling List, Scrollbars, Panels, Frames **Layouts:** Flow Layout, Grid Layout, Border Layout, Card Layout.	12

REFERENCE BOOKS:

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

Core Java Practical

COURSE CODE: BITS4P1 **COURSE CREDIT: 02**

Course Objectives:

- To teach basic and Object-Oriented programming concept using basic syntaxes of control Structures, strings and function for developing skills of logic building activity.
- To help students to learn AWT and Applet packages for effective GUI creation and Event handling capabilities.

List of Practical	
1.	**Java Basics**
a.	Write a Java program that takes a number as input and prints its multiplicationtable upto 10.
b.	Write a Java program to display the following pattern. ***** **** *** ** *
c.	Write a Java program to print the area and perimeter of a circle.
2.	**Use of Operators**
a.	Write a Java program to add two binary numbers.
b.	Write a Java program to convert a decimal number to binary number and viceversa.
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 SortData that contains the method asec() and desc().
b.	Designed a class that demonstrates the use of constructor and destructor.
c.	Write a java program to demonstrate the implementation of abstract class.
5.	**Inheritance**
a.	Write a java program to implement single level inheritance.

b.	Write a java program to implement method overriding
c.	Write a java program to implement multiple inheritance.
6.	**Packages and Arrays**
a.	Create a package, Add the necessary classes and import the package in java class.
b.	Write a java program to add two matrices and print the resultant matrix.
c.	Write a java program for multiplying two matrices and print the product for thesame.
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 programto 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.

REFERENCE BOOKS:

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

Introduction to Embedded Systems

COURSE CODE: BITS402 **COURSE CREDIT: 02**

Course Objectives:

- To acquire knowledge about the basic working of a microcontroller system and its programming using high level languages.
- To provide experiential learning to integrate hardware and software for microcontroller application systems.

Sr. No	Modules/Units	No of Lectures
1.	**Introduction:** Embedded Systems and general purpose computer systems, history, classifications, applications and purpose of embedded systems **Core of embedded systems:** microprocessors and microcontrollers, RISC and CISC controllers, Big endian and Little endian processors, Application specific ICs, Programmable logic devices, COTS, sensors and actuators, communication interface, embedded firmware, other system components. **Characteristics and quality attributes of embedded systems:** Characteristics, operational and non-operational quality attributes.	12
2.	**Embedded Systems – Application and Domain Specific:** Application specific – washing machine, domain specific - automotive. **Embedded Hardware:** Memory map, i/o map, interrupt map, processor family, external peripherals, memory – RAM , ROM, types of RAM and ROM, memory testing, CRC ,Flash memory. **Peripherals:** Control and Status Registers, Device Driver, Timer Driver - Watchdog Timers.	12
3.	**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
4.	**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
5.	**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	12

	Environment – IDE, types of file generated on cross compilation, disassembler/ de-compiler, simulator, emulator and debugging, embedded product development life-cycle, trends in embedded industry.	

REFERENCE BOOKS:

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

Introduction to Embedded Systems Practical

COURSE CODE: BITS4P2 **COURSE CREDIT: 02**

Course Objectives:

- Students will be provided with the knowledge to understand the Embedded systems design, Embedded programming and their operating system
- To make students learn embedded C programming in a microcontroller.

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

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

Applied Mathematics

COURSE CODE: BITS403 **COURSE CREDIT: 02**

Course Objectives:

- Students will be taught the basic concepts of matrices and complex numbers.
- Students will be taught to solve linear and higher order differential equations.
- Students will be taught the concepts of Laplace and inverse Laplace transform and solve differential equations by using Laplace and inverse Laplace transform.

Sr. No	Modules/Units	No of Lectures
1.	**Matrices:** Introduction, Types of Matrices, Determinant, Transpose of a Matrix, Conjugate of a Matrix, Transposed Conjugate of a Matrix, Operations of Matrices and Properties, Elementary Transformation, Inverse of a Matrix, Rank of a Matrix, Echelon or Normal Form of a Matrix, Linear Equations, Linear Dependence andLinear Independence of Vectors, Linear Transformation, Characteristics Roots and Characteristics Vectors, Properties of Characteristic Roots and Characteristic vectors, Caley-Hamilton Theorem, Similarity of Matrices, Reduction of a Matrix to a Diagonal Matrix which has Elements as Characteristics Values. **Complex Numbers:** Introduction, Equality of Complex Numbers, Graphical Representation of Complex Number (Argand's Diagram),Polar Form of Complex Numbers, Polar Form of x + iy for DifferentSigns of x, y, Exponential Form of Complex Numbers, Mathematical Operation with Complex Numbers and their Representation on Argand's Diagram, Circular Functions of Complex Angles, Definition of Hyperbolic Function, Relations between Circular and Hyperbolic Functions.	12
2.	**Functions of Single Variable**: Limit, Continuity and Differentiability, Mean Value Theorems - Introduction, Rolle's Theorem, Lagrange's Mean Value Theorem, Cauchy's Mean ValueTheorem (Statements and Applications), Increasing and Decreasing Functions, Basics of Optimization - Maxima and Minima.	12
3	**Differential Equations of the First Order and of the First Degree:** Introduction, Order and Degree of a Differential Equations, Separation of Variables, Equations Homogeneous in x and y, Non-Homogeneous Linear Equations, Exact Differential Equation, Integrating Factor, Linear Differential Equations and Differential Equations Reducible to this form, Method of Substitution. **Differential Equations of the First Order of a Degree Higher than the First:** Introduction, Solvable for p (or the method of factors), Solve for y, Solve for x, Clairaut's Form of the Equation, Methods of Substitution, Method of Substitution.	12

	. **Linear Differential Equations with Constant Coefficients:** Introduction, The Differential Operator, Linear Differential Equationf(D) y = 0, Different Cases Depending on the Nature of the Root of the Equation f(D) = 0, Linear Differential Equation f(D) y = X, The Complimentary Function, The Inverse Operator 1/f(D) and the Symbolic Expiration for the Particular Integral 1/f(D) X; The General Methods, Particular Integral : Short Methods, Particular Integral : Other Methods, Differential Equations Reducible to the Linear Differential Equations with Constant Coefficients.	
4.	**The Laplace Transform:** Introduction, Definition of the Laplace Transform, Table of Elementary Laplace Transforms, Theorems on Important Properties of Laplace Transformation, First Shifting Theorem, Second Shifting Theorem, The Convolution Theorem, Laplace Transform of an Integral, Laplace Transform of Derivatives,**Inverse Laplace Transform:** Shifting Theorem, Partial fraction Methods, Use of Convolution Theorem, Solution of Ordinary LinearDifferential Equations with Constant Coefficients, Solution of Simultaneous Ordinary Differential Equations, Laplace Transformation of Special Function, Periodic Functions, Heaviside Unit Step Function, Dirac-delta Function(Unit Impulse Function).	12
5.	**Beta and Gamma Functions:** Definitions**,** Properties and Problems. Duplication formula. **Differentiation Under the Integral SignError Functions**	12

REFERENCE BOOKS:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A Textbook of Applied Mathematics Vol I	P. N. Wartikar and J. N. Wartikar	Vidhyarthi Graha,	1st	2010
2.	A Textbook of Applied Mathematics Vol II	P. N. Wartikar and J. N. Wartikar	Vidhyarthi Graha,	1st	2010
3.	Higher Engineering Mathematics	Dr. B. S. Grewal	Khanna Publication	2nd	2012

Applied Mathematics Practical

COURSE CODE: BITS4P3 **COURSE CREDIT: 02**

Course Objectives:

- Students will get more problem solving sessions.

List of Practical:	
1.	Problems on Inverse by Adjoint Formla, Rank of a Matrix, Linear Equations and Concept of Linear Dependence and Independence.
2.	Problems on Eigen Values, Eigen Vectors and Cayley-Hamilton Theorem.
3.	Problems on Complex Numbers.
4.	Problems on Mean Value Theorems and Optimization.
5.	Problems on Differential Equations of First order and First Degree, and Differential Equations of the First Order of a Degree Higher than the First.
6.	Problems on Linear Differential Equations with Constant Coefficients.
7.	Problems on Laplace Transform.
8.	Problems on Inverse Laplace Transform.
9.	Problems on Beta and Gamma Functions.
10.	Problems on Differentiation Under the Integral Sign and Error Functions.

Software Engineering

COURSE CODE: BITS404 **COURSE CREDIT: 02**

Course Objectives:

- Students will be provided with the knowledge of basic Software engineering methods and practices, and their appropriate application.
- Students will be taught software engineering layered technology and Process framework.
- Students will be given a general understanding of software process models such as the waterfall and evolutionary models.
- To make the students understand software requirements and the SRS documents.

Sr. No	Modules/Units	No of Lectures
1.	**Introduction:** The Changing Nature of Software: WebApps, Mobile Applications, Cloud Computing, Product Line Software. What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc. **Software Requirements:** Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements. **Software Processes:** Process and Project, Component Software Processes. **Software Development Process Models**. • Waterfall Model. • Prototyping. • Iterative Development. • Rational Unified Process. • The RAD Model • Time boxing Model. **Agile software development:** Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.	12
2.	**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, Requirements elicitation and analysis, Requirements Validations,	12

	Requirements Management. **System Models:** Models and its types, Context Models, Behavioural Models, Data Models, Object Models, Structured Methods.	
3.	**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
4.	**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. **The Software Team, Team Structures.**	12
5.	**Process Improvement:** Process and product quality, Process Classification, Process Measurement, Process Analysis and Modelling, 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

REFERENCE BOOKS:

Sr. No.	Title	Authors	Publisher	Edition	Year
1.	Software Engineering,edition	Ian Somerville	Pearson Education	9th	-
2.	Software Engineering	Pankaj Jalote	Narosa Publication	-	-
3.	Software engineering, a practitioner‟s approach	Roger Pressman	Tata McGraw Hill	7th	-

Software Engineering Practical

COURSE CODE: BITS4P4 **COURSE CREDIT: 02**

Course Objectives:

- To make students understand different UML diagrams.
- To enable students to draw UML diagrams for developing software.

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.

REFERENCE BOOKS:

Sr. No.	Title	Authors	Publisher	Edition	Year
1.	Object - Oriented Modeling and Design	Michael Blaha, James Rumbaugh	Pearson	-	2011
2.	Learning UML 2. 0	Kim Hamilton,Russ Miles	O'Reilly Media	-	2006
3.	The unified modeling language user guide	Grady Booch, JamesRumbaugh, Ivar Jacobson	AddisonWesley	-	2005
4.	UML A BeginnersGuide	Jason T. Roff	McGraw HillProfessional	-	2003

Computer Graphics and Animation

COURSE CODE: BITS405 **COURSE CREDIT: 02**

Course Objectives:

- To make students learn the use of components of graphics system.
- To make students learn to convert the basic geometrical primitives and transform the shapes to fit them as per the picture definition.
- To make students comprehend and analyze the fundamentals of animation.

Sr. No	Modules/Units	No of Lectures
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. **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	12
2.	**Two-Dimensional Transformations:** Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations. **Three-Dimensional Transformations:**	12

	Three-Dimensional Scaling, Three-Dimensional Shearing, ThreeDimensional Rotation, Three-Dimensional Reflection, ThreeDimensional 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.	
3.	**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. **Light:** Radiometry, Transport, Equation, Photometry **Color:** Colorimetry, Color Spaces, Chromatic Adaptation, Color Appearance. **Polygon filling :** Edge –fill , Fence –fill, & Edge –flag polygon filling algorithms; simple Seed –fill & Scan –line seed –fill algorithms.	12
4.	**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. **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.	12
5.	**Computer Animation:** Principles of Animation, Key framing, Deformations, Character Animation, Physics-Based Animation, Procedural Techniques, Groups of Objects.	12

	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. **Introduction to Augmented Reality & Virtual Reality:** What is Augmented Reality? - Introduction, Where did Augmented Reality Come From? Augmented Reality. Augmented Reality Concepts - Introduction, How does Augmented Reality work? Concepts related to Augmented Reality.	

REFERENCE BOOKS:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Computer Graphics - Principles and Practice	J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes	Pearson	2nd	-
2.	Steve Marschner, Peter Shirley	Fundamentals of Computer Graphics	CRC press	4th	2016
3.	Computer Graphics	Hearn, Baker	Pearson	2nd	-
4.	Principles of Interactive Computer Graphics	William M. Newman and Robert F. Sproull	TMH	2nd	-
5.	Mathematical Elements for CG	D. F. Rogers, J. A. Adams	TMH	2nd	-
6.	Understanding Augmented Reality, Concepts and Applications	Alan B. Craig	Morgan Kaufmann Publishers	-	2013

Computer Graphics and Animation Practical

COURSE CODE: BITS4P5 **COURSE CREDIT: 02**

Course Objectives:

- Students will be able to write programs in C and C++ using graphics software.
- To make students learn screen coordinates and their pixels values using screen axis and design different shapes on screen using real world object coordinates.

List of Practical	
1.	**Solve the following:**
a.	Study and enlist the basic functions used for graphics in C / C++ / Python language. Give an example for each of them.
b.	Draw a co-ordinate axis at the center of the screen.
2.	**Solve the following:**
a.	Divide your screen into four region, draw circle, rectangle, ellipse and half ellipse in each region with appropriate message.
b.	Draw a simple hut on the screen.
3.	**Draw the following basic shapes in the center of the screen :**
	i. Circle ii. Rectangle iii. Square iv. Concentric Circles v. Ellipse vi. Line
4.	**Solve the following:**
a.	Develop the program for DDA Line drawing algorithm.
b.	Develop the program for Bresenham‟s Line drawing algorithm.
5.	**Solve the following:**
a.	Develop the program for the mid-point circle drawing algorithm.
b.	Develop the program for the mid-point ellipse drawing algorithm.
6.	**Solve the following:**
a.	Write a program to implement 2D scaling.
b.	Write a program to perform 2D translation
7.	**Solve the following:**
a.	Perform 2D Rotation on a given object.

b.	Program to create a house like figure and perform the following operations. i. Scaling about the origin followed by translation. ii. Scaling with reference to an arbitrary point. iii. Reflect about the line y = mx + c.
8.	**Solve the following:**
a.	Write a program to implement Cohen-Sutherland clipping.
b.	Write a program to implement Liang - Barsky Line Clipping Algorithm
9.	**Solve the following:**
a.	Write a program to fill a circle using Flood Fill Algorithm.
b.	Write a program to fill a circle using Boundary Fill Algorithm.
10.	**Solve the following:**
a.	Develop a simple text screen saver using graphics functions.
b.	Perform smiling face animation using graphic functions.
c.	Draw the moving car on the screen.
11.	**Solve the following:**
a.	Implementation of curve generation using Interpolation methods.
b.	Implementation of Curve generation using B-spline and Bezier curves.
c.	Implementation of any one of back face removal algorithm (such that depth buffer algorithm, Painter,,s algorithm, Warnock,,s algorithm, Scan line algorithm)

REFERENCE BOOKS:

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