



**SIES**

**College of Arts,  
Science &  
Commerce**

**RISE WITH EDUCATION**

**Sion (West), Mumbai – 400022.**

**(Autonomous)**

**Faculty: Science**

**Program: B.Sc.**

**Subject: INFORMATION TECHNOLOGY**

**Academic Year: 2022 – 2023**

**S.Y.B.Sc.**

**Credit Based Semester and Grading Syllabi approved  
by Board of Studies in Information Technology to be  
brought into effect from June 2022.**

### Semester III

<b>Course Code</b>	<b>Course Type</b>	<b>Course Title</b>	<b>Credits</b>
SIUSIT31	Skill Enhancement Course	Python Programming	2
SIUSIT32	Core Subject	Data Structures	2
SIUSIT33	Core Subject	Computer Networks	2
SIUSIT34	Core Subject	Database Management Systems	2
SIUSIT35	Core Subject	Applied Mathematics	2
SIUSITP31	Skill Enhancement Course Practical	Python Programming Practical	2
SIUSITP32	Core Subject	Data Structures Practical	2
SIUSITP33	Core Subject	Computer Networks Practical	2
SIUSITP34	Core Subject	Database Management Systems Practical	2
SIUSITP35	Core Subject	Mobile Programming Practical	2
<b>TOTAL CREDITS</b>			<b>20</b>

## Semester III

### Python Programming

#### Course Objective:

Logical thinking for software development is been introduced in First Year and Python programming is another platform to apply the same in a more complex way.

#### Course Outcome:

CO1: Write programs using the conditional statements and loops in Python and explain the importance of functions and apply various operations on strings.

CO2: Classify lists, tuples, dictionaries, and use files and Exceptions in Python.

CO3: Apply regular expression concepts for pattern matching and use various modules in Python and explain the complex data type Class.

CO4: Illustrate how MySQL database can be hooked up with Python code and used , also can develop basic GUI using widgets.

#### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSIT31</b>
<b>Course Name</b>	<b>Python Programming</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<b>Variables and Expressions</b> Values and Types, Variables, Variable Names and Keywords, Type conversion, Operators and Operands, Expressions, Interactive Mode and Script Mode, Order of Operations. <b>Conditional Statements:</b> if, if-else, nested if –else <b>Looping:</b> for, while, nested loops <b>Control statements:</b> Terminating loops, skipping specific conditions.	<b>12</b>
<b>II</b>	<b>Functions:</b> 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 <b>Strings:</b> 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..	<b>12</b>

<b>III</b>	<p><b>Lists:</b> 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</p> <p><b>Tuples and Dictionaries:</b> 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</p> <p>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</p> <p><b>Files:</b> Text Files, The File Object Attributes, Directories</p> <p><b>Exceptions:</b> Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions</p>	<b>12</b>
<b>IV</b>	<p><b>Regular Expressions</b> – Concept of regular expression, various types of regular expressions, using match function.</p> <p><b>Classes and Objects:</b> 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</p> <p><b>Modules:</b> Importing module, Creating and exploring modules, Math module, Random module, Time module</p>	<b>12</b>
<b>V</b>	<p><b>Creating the GUI Form and Adding Widgets:</b></p> <p><b>Widgets:</b> 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.</p> <p><b>Layout Management:</b> Designing GUI applications with proper Layout Management features.</p> <p><b>Look and Feel Customization:</b> Enhancing Look and Feel of GUI using different appearances of widgets.</p> <p><b>Storing Data in Our MySQL Database via Our GUI :</b> 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.</p>	<b>12</b>

### Books and References

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Think Python	Allen Downey	O'Reilly	1 <sup>st</sup>	2012
2	An Introduction to Computer Science using Python 3	Jason Montojo, Jennifer Campbell, Paul Gries	SPD	1 <sup>st</sup>	2014
3	Python GUI Programming Cookbook	Burkhard A. Meier	Packt		2015

4	Introduction to Problem Solving with Python	E. Balagurusamy	TMH	1 <sup>st</sup>	2016
5	Murach's Python programming	Joel Murach, Michael Urban	SPD	1 <sup>st</sup>	2017
6	Object-oriented Programming in Python	Michael H. Goldwasser, David Letscher	Pearson Prentice Hall	1 <sup>st</sup>	2008
7.	Exploring Python	Budd	TMH	1 <sup>st</sup>	2016

### Internal Evaluation: 40 Marks

20 Marks	20 Marks
Class Test	Design 2 forms (with at least 10 fields each) to accept data at the front end and store the same in the database (2 tables) at the back end and retrieve the data to the front end. The second form's data should add on information to the data in the first.

### Practical Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSITP31</b>
<b>Course Name</b>	<b>Python Programming Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

### List of Practical (Using any python IDE)

<b>1.</b>	<b>Write Python program to do the following: Use Function</b>
a.	Accept the name and age of the user. Print a message addressing the user the year they will turn 100 years old. For eg. If the name and age is entered as <b>Nimith</b> and <b>20</b> , the o/p will be <b>“Nimith, you will turn 100 in the year 2102”</b>
b.	Accept 2 numbers from the user and swap the numbers, with and without using multiple assignment statement. For eg. <b>If a=10 and b=20 before swapping then the o/p will be a=20 and b=10 after swapping.</b>
c.	Generate Fibonacci series with 10 terms, with and without using multiple assignment statement. Output will be <b>0 1 1 2 3 5 8 13 21 34</b>
d.	Accept a number and Reverse the number, also check if it is a Palindrome, use return statement. For eg. <b>If the i/p is 12345 then output will be 54321.</b> Eg. for Palindrome say for i/p <b>1221</b> o/p is <b>“1221 is a Palindrome”</b> , for i/p <b>1234</b> o/p <b>“1234 is not a Palindrome”</b>
e.	Accept a number and check if the given number is an Armstrong number or not. <b>For eg. 153 = 1<sup>3</sup> + 5<sup>3</sup> + 3<sup>3</sup></b>
f.	Accept a number and check if the given number is a Strong number or not. <b>For eg. 145 = 1! + 4! + 5!</b>
g.	Accept a number. Write a recursive function to print the factorial of the number. <b>For eg. 3! = 6</b>

<b>2.</b>	<b>Write Python program to do the following: Use String</b>
a.	Accept a string. Remove the characters at the odd index and print the original string and the transformed string. For eg. If original string S = “SIESASCS” the transformed string will be “SEAC”
b.	Accept a list of words and return the longest word and the length of the longest word. For eg. given a list L = [“Truth”, “Determination”, ”Perseverance”, “Freedom”, “Faithful”, “Courage”, “Hope”] the o/p will be “Longest word is “Determination” and length is 13”.
c.	Accept a comma separated sequence of words as input and print the words in sorted form (alphanumerically). For eg. i/p may be given as <b>Truth, Determination, Perseverance, Freedom,</b> and the o/p will be <b>Determination, Freedom, Perseverance, Truth</b>
d.	Count occurrences of a substring in a string. For eg. Given a string " <b>Fear leads to anger; anger leads to hatred; hatred leads to conflict; conflict leads to suffering.</b> ", if requested to find the occurrences of the substring “ <b>hatred</b> ” the o/p should be <b>2</b> .
e.	Reverse words in a string. For eg. Given a string “Don’t take rest after your first victory because if you fail in second, more lips are waiting to say that your first victory was just luck.” The o/p will be luck just was victory first your that say to waiting are lips more second, in fail you if because victory first your after rest take Don't
f.	A <b>pangram</b> is a sentence that contains all the letters of the English alphabet at least once, for example: <b>The quick brown fox jumps over the lazy dog.</b> Write a program to check whether the <b>given sentence is a pangram or not.</b>
g.	Count repeated characters in a string. For eg. Given a string “ <b>I have stood on a mountain of no’s for one yes</b> ” the o/p should be <b>10</b> <b>o 8</b> <b>n 5</b> <b>a 3</b> <b>e 3</b> <b>s 3</b> <b>t 2</b> <b>f 2</b>
h.	Convert a given string into a list of words. For eg. Given a string " <b>If there is no struggle, there is no progress.</b> " The o/p will be ['If', 'there', 'is', 'no', 'struggle,', 'there', 'is', 'no', 'progress.']
i.	Count and display the count of vowels and the vowels in a given text. For eg. Given a string “ <b>When one door of happiness closes, another opens; but often we look so long at the closed door that we do not see the one which has been opened for us.</b> ” The o/p should be count of vowels is <b>48</b> and the vowels are [e', 'o', 'e', 'o', 'o', 'o', 'a', 'i', 'e', 'o', 'e', 'a', 'o', 'e', 'o', 'e', 'u', 'o', 'e', 'e', 'o', 'o', 'o', 'o', 'a', 'e', 'o', 'e', 'o', 'o', 'a', 'e', 'o', 'o', 'e', 'e', 'e', 'o', 'e', 'i', 'a', 'e', 'e', 'o', 'e', 'e', 'o', 'u']
j.	Remove spaces from a given string. For eg. Given a string " <b>Do one thing every day that scares you.</b> " The o/p will be <b>Doonethingeverydaythatscaresyou.</b>
<b>3.</b>	<b>Write the program to do the following: Use List</b>
a.	Write a program that takes two lists and returns <b>True</b> if they have at least one common member and returns <b>None</b> if no common member is found. For eg. Given the 2 lists [1,2,3,4,5], [5,6,7,8,9] o/p will be <b>True</b> , given i/p [1,2,3,4,5], [6,7,8,9] the o/p will be <b>None</b>

b.	Write a program to print a specified list after removing the 1st, 2nd, 5th elements. For eg. Given the list ['Action', 'Believe', 'Commitment', 'Confidence', 'Dare', 'Focus'] o/p will be ['Action', 'Commitment', 'Dare']
c.	Accept a list from the user. Write a program to get the largest and smallest number in the list.
d.	Write a program to count the number of strings where the string length is 2 or more and the first and last character are same from a given list of strings. For eg. Given the list ['a','charismatic', 'champion', 'beaming', '4554','dignified'] the o/p will be 3
e.	Write a program to shuffle and print a specified list. For eg. Given the list ['Action', 'Believe', 'Commitment', 'Confidence', 'Dare', 'Focus'] o/p will be ['Commitment', 'Focus', 'Action', 'Believe', 'Dare', 'Confidence']
f.	Write a program to select an item randomly from a list. Given the list ['Action', 'Believe', 'Commitment', 'Confidence', 'Dare', 'Focus'] o/p will be ['Commitment'] (Note :user has no control over the choice)
<b>4. Write the program to do the following: Use Tuple</b>	
a.	Write a program to convert the given list to a tuple. For eg. Given the list [45, 23, 67, 12, 7, 99] o/p will be (45, 23, 67, 12, 7, 99).
b.	Write a program to calculate the product of all the numbers in a given tuple. For eg. Given the tuple (45, 23, 22, 2, -16, 19) the o/p will be -13844160
c.	Print all pair combinations of given 2 tuples. For eg. Given tuple 1 : (1, 3) and tuple 2 : (4, 9) the o/p , the combined tuple will be : [(1, 4), (1, 9), (3, 4), (3, 9), (4, 1), (4, 3), (9, 1), (9, 3)]
d.	Test if tuple is distinct. For eg. Given tuple : (13, 54, 95, 76, 11, 54) o/p will be False, given tuple : (13, 54, 95, 76, 11, 584) o/p will be True
<b>5. iii. Write the program to do the following: Use Dictionary</b>	
a.	Write a Python script to sort (ascending and descending) a dictionary by key and by value. For eg. Given the dictionary  <b>{1:2, 3:4, 4:3, 2:0, 0:0}</b>  o/p will be  <b>Original Dictionary : {1: 2, 3: 4, 4: 3, 2: 0, 0: 0}</b>  <b>Ascending order by Value : {2: 0, 0: 0, 1: 2, 4: 3, 3: 4}</b> <b>Descending order by Value : {3: 4, 4: 3, 1: 2, 2: 0, 0: 0}</b>  <b>Ascending order by Key : {0: 0, 1: 2, 2: 0, 3: 4, 4: 3}</b> <b>Descending order by Key : {4: 3, 3: 4, 2: 0, 1: 2, 0: 0}</b>
b.	Create grade calculator. For eg. Given the dictionaries <b># 1. Trusha's dictionary</b> <b>trusha = { "name":"Trusha Salian",</b> <b>          "assignment" : [90, 90, 90, 90],</b> <b>          "test" : [95, 95],</b> <b>          "lab" : [90.10, 90.10]}</b>  <b># 2. Chrishanth's dictionary</b>

	<pre>chrishanth = { "name": "Chrishant Lukshmanraj",                "assignment" : [82, 76, 48, 50],                "test" : [90, 90],                "lab" : [66.30, 66.42]              }</pre> <p>o/p will be</p> <p><b>Trusha Salian</b> ----- <b>Average marks of Trusha Salian is : 93.52</b> <b>Grade of Trusha Salian is : A</b></p> <p><b>Chrishant Lukshmanraj</b> ----- <b>Average marks of Chrishant Lukshmanraj is : 82.672</b> <b>Grade of Chrishant Lukshmanraj is : B</b></p>
c.	Write a Python script to print a dictionary where the keys are numbers between 1 and 15 (both included) and the values are square of keys. The o/p will be {1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100, 11: 121, 12: 144, 13: 169, 14: 196, 15: 225}
<b>6.</b>	<b>Write the program to do the following: Use File</b>
a.	Write a program to read an entire text file.
b.	Write a program to append text to a file and display the text.
c.	Write a program to read last n lines of a file.
<b>7.</b>	<b>Write the program to do the following: Use Class</b>
a.	<p>Create a class named Parent with constructor to initialise firstname and lastname and a method to display the same.</p> <p>Derive a class Child from Parent, add a constructor here to initialise the age and Adhaar Card number and a method to display the same.</p> <p>Derive a class GrandChild from Child, add a constructor here to initialise the address and PAN Card number and a method to display the same.</p> <p>Write a driver code to implement Multilevel Inheritance.</p>
b.	A polygon is a closed figure with 3 or more sides. Create a class called Polygon with data attributes to store the number of sides n and magnitude of each side as a list called sides. Add 2 methods, inputSides() method that takes in the magnitude of each side and a method dispSides() displays these side lengths. A square is a polygon with 4 sides. Create a class called square which inherits from Polygon. Define a method findAreaofsquare() to find and print the area of the square. Write a driver code to implement the above.
<b>8.</b>	<b>Write the program to do the following:</b>
a.	<p>Create a user defined module <b>stropersodfib</b> . Import the same in a Python (.py) code and use the functions defined in that user defined module and for the following:</p> <p>i) Check for Prime number</p> <p>ii) Check for Perfect number</p> <p>iii) Find the sum of the digits of a given number</p>



	iv) Print the Series 1, 20, 400, 8000, 160000 up to 10 terms. Use user defined modules.
b.	Accept the string <b>“We God’s creation! Worried about imperfection. Why lot of confusion? Which ends up in tension! Unable to pay attention? You are the one in control of the situation. Find a solution.”</b> as data. Print the words ending in “tion”, Print the words starting with ‘W or a’ and count the same, Count the number of lines in the string, Count the number of words ending in “tion” and not ending in “tion”. Use regular expressions.
c.	Write a program to implement exception handling. Demonstrate use of <b>ZeroDivisionError</b> and a user defined exception ( a variable is uninitialized and is been used in the code , this should raise an exception say <b>‘Variable not initialize’</b> ).
<b>9.</b>	<b>Write the program for the following: Use Widget</b>
a.	Demonstrate the use of the different Widgets say Label, Button, ComBox, CheckButton, RadioButton, Entry, Frame, Message in Python tkinter.
<b>10.</b>	<b>Design the database applications for the following:</b>
a.	Create a database Student in <b>SQL Server</b> .  Create a table Sdetails with columns Sid, Sname, Sage, Spnrnumber and Sdept . Sid is the Primary key of the table.  Create a table Department with columns Deptid and Deptname. Deptid is the Primary key of the table.  Sdept column in Sdetails table is the Foreign key whose values are derived from the Primary key column Deptid in Department table.  Insert values into both the tables.  Update values in the rows of the tables. Delete rows in the tables.
<b>11.</b>	<b>Programs based on data analytics and automation.</b>

**Books and References:**

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

## Data Structures

### Course Objective:

To develop sound techniques on designing, developing, and documenting well-structured programs using proper methods and continue to apply problem solving skills and provide a foundation for advanced programming courses using an OOP (object-oriented programming) methodology.

### Course Outcome:

CO1: Identify the need of different data structures and choose appropriate data structures to represent data items in real world problem.

CO2: Analyse time and space complexities of the algorithms

CO3: Design programs using various data structures such as arrays, linked list, stack, queues, heap, graphs, binary trees, B-trees.

CO4: Analyse and implement various kinds of searching and sorting techniques.

### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSIT32</b>
<b>Course Name</b>	<b>Data Structures</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<b>Introduction:</b> 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. <b>Array:</b> 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 Representation of Special kind of Matrices, Advantages and Limitations of Arrays.	<b>12</b>
<b>II</b>	<b>Linked List:</b> 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	<b>12</b>

	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.	
<b>III</b>	<p><b>Stack:</b> 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.</p> <p><b>Queue:</b> 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.</p>	<b>12</b>
<b>IV</b>	<p><b>Sorting and Searching Techniques</b> Bubble, Selection, Insertion, Merge Sort. Searching: Sequential, Binary, Indexed Sequential Searches, Binary Search.</p> <p><b>Tree:</b> 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.</p> <p><b>Advanced Tree Structures:</b> Red Black Tree, Operations Performed on Red Black Tree, AVL Tree, Operations performed on AVL Tree, 23 Tree, B-Tree.</p>	<b>12</b>
<b>V</b>	<p><b>Hashing Techniques</b> Hash function, Address calculation techniques, Common hashing functions Collision resolution, Linear probing, Quadratic, Double hashing, Bucket hashing, Deletion and rehashing</p> <p><b>Graph:</b> 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.</p>	<b>12</b>

### Books and References

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A Simplified Approach to Data Structures	Lalit Goyal, Vishal Goyal, Pawan Kumar	SPD	1 <sup>st</sup>	2014
2.	An Introduction to Data Structure with Applications	Jean – Paul Tremblay and Paul Sorenson	Tata MacGraw Hill	2 <sup>nd</sup>	2007
3.	Data Structure and Algorithm	Maria Rukadikar	SPD	1 <sup>st</sup>	2017
4.	Schaum's Outlines Data structure	Seymour Lipschutz	Tata Mc Graw Hill	2 <sup>nd</sup>	2005

5.	Data structure – A Pseudocode Approach with C	AM Tanenbaum, Y Langsam and MJ Augustein	Prentice Hall India	2 <sup>nd</sup>	2006
6.	Data structure and Algorithm Analysis in C	Weiss, Mark Allen	Addison Wesley	1 <sup>st</sup>	2006

### Internal Evaluation: 40 Marks

<b>20 Marks</b>	<b>20 Marks</b>
Class Test	Assignments(programs)

### Practical Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSITP32</b>
<b>Course Name</b>	<b>Data Structures Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

### List of Practical: (Using Visual Studio with C++)

<b>1.</b>	<b>Implement the following:</b>
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]
<b>2.</b>	<b>Implement the following for Linked List:</b>
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.
<b>3.</b>	<b>Implement the following for Stack:</b>
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.
<b>4.</b>	<b>Implement the following for Queue:</b>
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.
<b>5.</b>	<b>Implement the following sorting techniques:</b>
a.	Write a program to implement bubble sort.
b.	Write a program to implement selection sort.
c.	Write a program to implement insertion sort.
<b>6.</b>	<b>Implement the following data structure techniques:</b>
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.
<b>7.</b>	<b>Implement the following data structure techniques:</b>
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
<b>8.</b>	<b>Implement the following data structure techniques:</b>
a.	Write a program to insert the element into maximum heap.
b.	Write a program to insert the element into minimum heap.
<b>9.</b>	<b>Implement the following data structure techniques:</b>
a.	Write a program to implement the collision technique.
b.	Write a program to implement the concept of linear probing.
<b>10.</b>	<b>Implement the following data structure techniques:</b>
a.	Write a program to generate the adjacency matrix.
b.	Write a program for shortest path diagram.

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

## Computer Networks

### Course Objective:

To orient the students about the OSI networking model and study the bottom four layers of the model in detail.

### Course Outcome:

CO1: Analyze the requirements for a given organizational structure and select the most appropriate networking architecture and technologies.

CO2: Use networking protocols, and their hierarchical relationship in the context of a conceptual model, such as the OSI and TCP/IP framework.

CO3: Explain the OSI layers with their services and protocols.

### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSIT33</b>
<b>Course Name</b>	<b>Computer Networks</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<b>Introduction:</b> Data communications, networks, network types <b>Network Models:</b> Protocol layering, TCP/IP protocol suite, The OSI model. <b>Introduction to Physical layer:</b> Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance. <b>Digital and Analog transmission:</b> Digital-to-digital conversion, analog-to-digital conversion, transmission modes, digital-to-analog conversion, analog-to-analog conversion.	<b>12</b>
<b>II</b>	<b>Transmission media:</b> Guided Media, Unguided Media <b>Switching:</b> Introduction, circuit switched networks, packet switching <b>Introduction to the Data Link Layer:</b> Link layer addressing, Error detection and correction, block coding, cyclic codes, checksum	<b>12</b>
<b>III</b>	<b>Data Link Control:</b> DLC services, data link layer protocols, HDLC, Point-to-point protocol. <b>Media Access Control:</b> Random access, controlled access, channelization, Wired LANs – Ethernet Protocol, standard ethernet, fast ethernet, gigabit ethernet, 10 gigabit ethernet, <b>Connecting devices and Virtual LANs.</b>	<b>12</b>

<b>IV</b>	<p><b>Introduction to the Network Layer:</b> Network layer services, packet switching, network layer performance, IPv4 addressing, Internet Protocol, ICMPv4, Mobile IP</p> <p><b>Unicast Routing:</b> Introduction, routing algorithms, unicast routing protocols.</p> <p><b>Next generation IP:</b> IPv6 addressing, IPv6 protocol, ICMPv6 protocol, transition from IPv4 to IPv6.</p>	<b>12</b>
<b>V</b>	<p><b>Introduction to the Transport Layer:</b> 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,</p> <p><b>Standard Client Server Protocols:</b> World wide-web and HTTP, FTP, Electronic mail, Telnet, Secured Shell, Domain name system.</p>	<b>12</b>

### Books and References

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

### Internal Evaluation: 40 Marks

20 Marks	20 Marks
Class Test	Test various protocols on a given topology

### Practical Component

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSITP33</b>
<b>Course Name</b>	<b>Computer Networks Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

### List of Practical (Using network simulators)

<b>1.</b>	<p><b>IPv4 Addressing and Subnetting</b></p> <p>a) Given an IP address and network mask, determine other information about the IP address such as:</p> <ul style="list-style-type: none"> <li>• Network address</li> </ul>
-----------	--

	<ul style="list-style-type: none"> <li>• Network broadcast address</li> <li>• Total number of host bits</li> <li>• Number of hosts</li> </ul> <p>b) Given an IP address and network mask, determine other information about the IP address such as:</p> <ul style="list-style-type: none"> <li>• The subnet address of this subnet</li> <li>• The broadcast address of this subnet</li> <li>• The range of host addresses for this subnet</li> <li>• The maximum number of subnets for this subnet mask</li> <li>• The number of hosts for each subnet</li> <li>• The number of subnet bits</li> <li>• The number of this subnet</li> </ul>
<b>2.</b>	Use of ping and tracert / traceroute, ipconfig / ifconfig, route and arp utilities.
<b>3.</b>	Configure IP static routing.
<b>4.</b>	Configure IP routing using RIP.
<b>5.</b>	Configuring Simple OSPF.
<b>6.</b>	Configuring DHCP server and client.
<b>7.</b>	Create virtual PC based network using virtualization software and virtual NIC.
<b>8.</b>	Configuring DNS Server and client.
<b>9.</b>	Configuring OSPF with multiple areas.
<b>10.</b>	Use of Wireshark to scan and check the packet information of following protocols <ul style="list-style-type: none"> <li>• HTTP</li> <li>• ICMP</li> <li>• TCP</li> <li>• SMTP</li> <li>• POP3</li> </ul>



## Database Management Systems

### Course Objective:

To acquaint learners about the importance of data model in designing a database along with usage of SQL and PL/SQL.

### Course Outcome:

- CO1: Examine and conceptualize data using the relational model and create Entity Relationship diagrams for data models.  
CO2: Use SQL and PL/SQL to create, manage the database objects in the database ,retrieve data and program data in the database.  
CO3: Explain the ACID properties of transactions, different types scheduling in transactions, concurrency control and recovery management in DBMS.

### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSIT34</b>
<b>Course Name</b>	<b>Database Management Systems</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<b>Introduction to Databases and Transactions</b> What is database system, purpose of database system, view of data, relational databases, database architecture, transaction management <b>Data Models</b> The importance of data models, Basic building blocks, Business rules, The evolution of data models, Degrees of data abstraction.	<b>12</b>
<b>II</b>	<b>Database Design, ER Diagram</b> Database design and ER Model: overview, ER Model, Constraints, ER Diagrams, ERD Issues, weak entity sets, Relational Schemas <b>Relational database model:</b> Logical view of data, keys, integrity rules, Relational Database design: features of good relational database design, atomic domain and Normalization (1NF, 2NF, 3NF, BCNF) with cases.	<b>12</b>
<b>III</b>	<b>Transaction management and Concurrency</b> Control Transaction management: ACID properties, serializability and concurrency control, Lock based concurrency control (2PL, Deadlocks), Time stamping methods, optimistic methods, database recovery management. <b>SQL:</b> Introduction, Relational Databases, SQL Basics, <b>Simple Queries</b> : SELECT statement, WHERE clause, Search conditions in a WHERE clause, Multitable Queries, <b>Summary Queries:</b> Column functions, Grouped Queries, Group Search Conditions	<b>12</b>

<b>IV</b>	<p><b>SQL : Subqueries:</b> Using subqueries, Subquery search conditions, subqueries and joins, Nested subqueries, Correlated subqueries,</p> <p><b>Creating Database :</b> Create, Alter Drop statements ,Database Updates : Insert, Update ,Delete statements, Data Integrity, Views, SQL Security</p> <p><b>PL-SQL: Overview of PL/SQL:</b> About PL/SQL, Environment, Benefits, <b>Declaring Variables:</b> PL/SQL Block Structure, Block Types, Handling variables, Declaring variables, %TYPE, PUT_LINE procedure, <b>Writing Executable Statements:</b> PL/SQL Block syntax, Identifiers, Comments, Scope, Operators, <b>Interacting with Oracle Server :</b> SQL statements in PL/SQL, Naming Conventions, SQL Cursor and its attributes</p>	<b>12</b>
<b>V</b>	<p><b>PL-SQL: Writing Control Structures :</b> Conditional IF statements, CASE expressions, LOOP statements, <b>Working with Composite Datatypes:</b> PL/SQL Records, %ROWTYPE, INDEX BY Table ,INDEX BY Table of Records, <b>Writing Explicit Cursors :</b> Steps to create and use explicit cursors, Explicit cursor attributes, Cursors and Records, Cursor For Loops, <b>Handling Exceptions:</b> Pre-defined, User-defined , RAISE_APPLICATION_ERROR, <b>Procedures :</b> What is a procedure, syntax, modes, removing a procedure, <b>Functions:</b> What is a function, syntax, executing functions, removing a function, Comparing procedures and functions, Packages: What are packages, components, creating a package, removing a package, Advantages, <b>Triggers:</b> What is a trigger, types, guidelines for designing triggers, Statement and Row triggers, Creation and use of DML triggers. Using conditional Predicates, Using OLD and NEW Qualifiers, Removing a Trigger.</p>	<b>12</b>

### Books and References

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Database System and Concepts	A Silberschatz, H Korth, S Sudarshan	McGraw-Hill	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
5.	SQL – The Complete Reference	Paul Weinberg, James Groff, Andrew Oppel	McGraw-Hill	Third	2010
6.	Introduction to Oracle9i: PL/SQL:Volume I & II		Oracle		2001

**Internal Evaluation: 40 Marks**

<b>20 Marks</b>	<b>20 Marks</b>
Class Test	Design the ER model and the data dictionary for a real world scenario

**Practical Component:**

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSITP34</b>
<b>Course Name</b>	<b>Database Management Systems Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

**List of Practical: (Using Oracle Live SQL/SQL Server/MySQL)**

<b>1.</b>	<b>SQL Statements – 1</b>
a.	Writing Basic SQL SELECT Statements
b.	Restricting Data – WHERE clause
c.	Sorting Data – ORDER BY clause
<b>2.</b>	<b>SQL Statements – 2</b>
a.	Displaying Data from Multiple Tables – Equi joins, Self joins, Outer joins
b.	Aggregating Data Using Group Functions – Summary Functions and Group BY
c.	Subqueries
<b>3.</b>	<b>Creating and Managing Tables</b>
a.	Creating and Managing Tables : Use CREATE, ALTER DROP statements to create a set of tables that form a part of a database. (Handle Referential Cycles)
b.	Including Constraints : Primary key, foreign key, Not null, Check
<b>4.</b>	<b>Manipulating Data</b>
a.	Using INSERT, UPDATE, DELETE statement to insert rows, update rows and delete rows in the tables created in practical 3 (Handle referential cycles)
<b>5.</b>	<b>Creating and Managing other database objects</b>
a.	Creating Views: Create view , with check option
b.	Other Database Objects : Sequences, Synonyms, Index
<b>6.</b>	<b>Using SET operators, GROUP BY clause (advanced features), hierarchical retrieval</b>
a.	Using SET Operators
b.	Enhancements to the GROUP BY Clause like Roll Up, Cube, Grouping Sets.
c.	Hierarchical retrieval queries

<b>7.</b>	<b>PL/SQL Basics</b>
a.	Declaring Variables and Executing SQL statements in PL/SQL
b.	Writing Control Structures- Loops
<b>8.</b>	<b>Composite data types, cursors and exceptions.</b>
a.	Working with Composite Data Types – RECORD,INDEX BY
b.	Writing Explicit Cursors and CURSOR for loops
c.	Handling Exceptions
<b>9.</b>	<b>Procedures and Functions</b>
<b>10.</b>	<b>Packages and Database Triggers</b>

### Books and References:

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Database System and Concepts	A Silberschatz, H Korth, S Sudarshan	McGraw-Hill	Fifth Edition	
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

## Applied Mathematics

### Course Objective:

To develop fundamental mathematical skills and the ability for independent mathematical learning and reasoning.

### Course Outcome:

CO1: Apply mathematical concepts and principles like matrices, linear equations to perform computations

CO2: Solve problems based on complex numbers and linear differential equations, multiple integrals and apply the concepts of integration

CO3: Evaluate Laplace transforms and inverse Laplace transforms of various functions

### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSIT35</b>
<b>Course Name</b>	<b>Applied Mathematics</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<b>Matrices:</b> Inverse of a matrix, Properties of matrices, Elementary Transformation, Rank of Matrix, Echelon or Normal Matrix, Inverse of matrix, Linear equations, Linear dependence and linear independence of vectors, Linear transformation, Characteristics roots and characteristics vectors, Properties of characteristic vectors, Caley Hamilton Theorem, Similarity of matrices.	<b>12</b>
<b>II</b>	<b>Complex Numbers:</b> Complex number, Equality of complex numbers, Graphical representation of complex number(Argand's Diagram), Polar form of complex numbers, Polar form of $x+iy$ for different signs of $x,y$ , Exponential form of complex numbers, Mathematical operation with complex numbers and their representation on Argand's Diagram, Circular functions of complex angles, Definition of hyperbolic function, Relations between circular and hyperbolic functions, Inverse hyperbolic functions, Differentiation and Integration, Graphs of the hyperbolic functions, Logarithms of complex quality, $j(=i)$ as an operator(Electrical circuits)	<b>12</b>
<b>III</b>	<b>Equation of the first order and of the first degree:</b> Separation of variables, Equations homogeneous in $x$ and $y$ , Non-homogeneous linear equations, Exact differential Equation, Integrating Factor, Linear Equation and equation reducible to this form. <b>Linear Differential Equations with Constant Coefficients:</b> Introduction, The Differential Operator, Linear Differential Equation $f(D) y = 0$ ,	<b>12</b>

	Different cases depending on the nature of the root of the equation $f(D) = 0$ , Linear differential equation $f(D) y = X$ , The inverse operator $1/f(D)$ and the symbolic expansion for the particular integral $1/f(D) X$ ; the general methods	
<b>IV</b>	<b>The Laplace Transform:</b> 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 (only statement), Laplace Transform of an Integral, Laplace Transform of Derivatives, <b>Inverse Laplace Transform:</b> Shifting Theorem, Partial fraction Methods, Use of Convolution Theorem, Solution of Ordinary Linear Differential Equations with Constant Coefficients.	<b>12</b>
<b>V</b>	<b>Multiple Integrals:</b> Double Integral, Change of the order of the integration, Triple integrals. <b>Applications of integration:</b> Areas, Volumes of solids. <b>Beta and Gamma Functions</b> – Definitions, Properties and Problems. Duplication formula.(without proof) <b>Differentiation Under the Integral Sign</b> <b>Error Functions</b>	<b>12</b>

### Books and References

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

### Internal Evaluation: 40 Marks

<b>20 Marks</b>	<b>20 Marks</b>
Class Test	Assignments / Problem solving

**Practical Component:**

<b>B. Sc (Information Technology)</b>	<b>Semester – III - SIUSITP35</b>
<b>Course Name</b>	<b>Mobile Programming Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

**List of Practical: (Using Cordova,Flutter)**

<b>Setting up CORDOVA, Flutter Application</b>	
<b>1.</b>	<ul style="list-style-type: none"> <li>• Creating and building simple “Hello World” App using Cordova</li> <li>• Adding and Using Buttons</li> <li>• Adding and Using Event Listeners</li> </ul>
<b>2.</b>	<ul style="list-style-type: none"> <li>• Creating and Using Functions</li> <li>• Using Events</li> <li>• Handling and Using Back Button</li> </ul>
<b>3.</b>	<ul style="list-style-type: none"> <li>• Installing and Using Plugins</li> <li>• Installing and Using Battery Plugin</li> <li>• Installing and Using Camera Plugin</li> </ul>
<b>4.</b>	<ul style="list-style-type: none"> <li>• Installing and Using Contacts Plugin</li> <li>• Installing and Using Device Plugin</li> <li>• Installing and Using Accelerometer Plugin</li> </ul>
<b>5.</b>	<ul style="list-style-type: none"> <li>• Install and Using Device Orientation plugin</li> <li>• Create and Using Prompt Function</li> </ul>
<b>6.</b>	<ul style="list-style-type: none"> <li>• Installing and Using File Plugin</li> <li>• Installing and Using File Transfer Plugin</li> <li>• Using Download and Upload functions</li> </ul>
<b>7.</b>	<ul style="list-style-type: none"> <li>• Installing and Using Globalization Plugin</li> <li>• Installing and Using Media Plugin</li> <li>• Installing and Using Media Capture Plugin</li> </ul>
<b>8.</b>	<ul style="list-style-type: none"> <li>• Installing and Using Network Information Plugin</li> <li>• Installing and Using Splash Screen Plugin</li> <li>• Installing and Using Vibration Plugin</li> </ul>
<b>9.</b>	<ul style="list-style-type: none"> <li>• Developing Single Page Apps</li> <li>• Developing Multipage Apps</li> <li>• Storing Data Locally in a Cordova App</li> </ul>

<b>10.</b>	<ul style="list-style-type: none"> <li>• Use of sqlite plugin with PhoneGap / apache Cordova</li> <li>• Using Sqlite read/write and search</li> <li>• Populating Cordova SQLite storage with the JQuery API</li> </ul>
------------	--

**Books and References:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Beginning Flutter: A hands on Guide to App Development	Marco L. Napoli	John Wily & Sons	Copyrighted	2020
2.	Apache Cordova in Action	Raymond Camden	Manning Publications	1 <sup>st</sup>	2015
3.	PhoneGap By Example	Andrey Kovalenko	PACKT Publishing	1 <sup>st</sup>	2015



## Semester IV

Course Code	Course Type	Course Title	Credits
SIUSIT41	Skill Enhancement Course	Core Java	2
SIUSIT42	Core Subject	Introduction to Embedded Systems and Internet of Things	2
SIUSIT43	Core Subject	Computer Oriented Statistical Techniques	2
SIUSIT44	Core Subject	Software Engineering	2
SIUSIT45	Core Subject	Computer Graphics and Animation	2
SIUSITP41	Skill Enhancement Course Practical	Core Java Practical	2
SIUSITP42	Core Subject Practical	Introduction to Embedded Systems and Internet of Things Practical	2
SIUSITP43	Core Subject Practical	Computer Oriented Statistical Techniques Practical	2
SIUSITP44	Core Subject Practical	Software Engineering Practical	2
SIUSITP45	Core Subject Practical	Computer Graphics and Animation Practical	2
<b>TOTAL CREDITS</b>			<b>20</b>

## Semester IV

### Core Java

#### **Course Objective:**

The learner is introduced to the fundamentals of Java programming. It includes basics and advanced features such as multithreaded programming, event handling and java swings.

#### **Course Outcome:**

CO1: Explain the features, data types and control flow statements used in Java programming language

CO2: Write java programs based on object oriented concepts like polymorphism, Inheritance and interfaces, packages.

CO3: Design Multiple threads, handle exceptions and use event handling and swings to develop GUI applications that suit user requirements.

#### **Theory Component:**

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSIT41</b>
<b>Course Name</b>	<b>Core Java</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<b>Introduction:</b> 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, Methods References, 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 <b>Data types:</b> 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.	<b>12</b>
<b>II</b>	<b>Control Flow Statements:</b> The If...Else If...Else Statement, The Switch...Case Statement <b>Iterations:</b> The While Loop, The Do ... While Loop, The For Loop, The Foreach Loop, Labeled Statements, The Break And Continue Statements, The Return Statement <b>Classes:</b> 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],	<b>12</b>

	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.	
<b>III</b>	<b>Inheritance:</b> Derived Class Objects, Inheritance and Access Control, Default Base Class Constructors, this and super keywords. Abstract Classes And Interfaces, Abstract Classes, Abstract Methods, Interfaces, What Is An Interface? How Is An Interface Different From An Abstract Class?, Multiple Inheritance, Default Implementation, Adding New Functionality, Method Implementation, Classes V/s Interfaces, Defining An Interface, Implementing Interfaces. <b>Packages:</b> Creating Packages, Default Package, Importing Packages, Using A Package.	<b>12</b>
<b>IV</b>	<b>Enumerations, Arrays:</b> 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. <b>Multithreading:</b> the thread control methods, thread life cycle, the main thread, creating a thread, extending the thread class. <b>Exceptions:</b> Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause	<b>12</b>
<b>V</b>	<b>Event Handling:</b> Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes and inner classes. <b>Swings:</b> Swing Overview, The MVC Architecture, <b>Core Swing Components:</b> JComponent, JToolTip, JLabel, JButton, JPanel, <b>Toggle Buttons:</b> JToggleButton, JCheckBox, JRadioButton, Swing Menus and Toolbars, <b>RootPane containers:</b> JRootPane, JFrame, JWindow, JDialog, <b>Pop-Ups and Choosers:</b> JOptionPane, JColorChooser, JFileChooser, <b>LayoutManagers:</b> FlowLayout, BorderLayout, GridLayout, CardLayout, BoxLayout, <b>Advanced Swing Containers:</b> JSplitPane, JTabbedPane, JScrollPane, <b>Bounded Range Components:</b> JScrollBar, JProgressBar, <b>List Controls:</b> JList, JComboBox, Basic text components, Trees, Tables	<b>12</b>

## Books and References

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

6.	Core Java: An Integrated Approach	R. Nageswara Rao	DreamTech	1st	2008
7.	The Definitive Guide to Java Swing	John Zukowski2	Apress	3rd	2005

### Internal Evaluation: 40 Marks

<b>20 Marks</b>	<b>20 Marks</b>
Class Test	Design Projects written in java using basic java concepts and GUI programming

### Practical Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSITP41</b>
<b>Course Name</b>	<b>Core Java Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

### List of Practical: (Using Notepad/NetBeans/Eclipse/BlueJ)

<b>1.</b>	<b>Java Basics</b>
a.	Write a Java program that takes a number as input and prints its multiplication table upto 10.
b.	Write a Java program to display the various patterns.
c.	Write a Java program to print the area and perimeter of a circle.
<b>2.</b>	<b>Use of Operators</b>
a.	Write a Java program to add two binary numbers.
b.	Write a Java program to convert a decimal number to binary number and vice versa.
c.	Write a Java program to reverse a string.
<b>3.</b>	<b>Java Data Types</b>
a.	Write a Java program to count the letters, spaces, numbers and other characters of an input string.
b.	Implement a Java function that calculates the sum of digits for a given char array consisting of the digits '0' to '9'. The function should return the digit sum as a long value.
c.	Find the smallest and largest element from the array
<b>4.</b>	<b>Methods and Constructors</b>
a.	Designed a class SortData that contains the method asc() 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.

<b>5.</b>	<b>Inheritance</b>
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.
<b>6.</b>	<b>Packages and Arrays</b>
a.	Create a package, Add the necessary classes and import the package in java class.
b.	Write a java program to add two matrices and print the resultant matrix.
c.	Write a java program for multiplying two matrices and print the product for the same.
<b>7.</b>	<b>Vectors and Multithreading</b>
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.
<b>8.</b>	<b>GUI Programming</b>
a.	Core Swing Components: JComponent, JToolTip, JLabel, JButton, JPanel
b.	RootPane containers: JRootPane, JFrame, JWindow, JDialog
<b>9.</b>	<b>GUI Programming</b>
a.	Pop-Ups and Choosers: JOptionPane, JColorChooser, JFileChooser
b.	LayoutManagers: FlowLayout, BorderLayout, GridLayout, CardLayout, BoxLayout
<b>10.</b>	<b>GUI Programming.</b>
a.	Advanced Swing Containers: JSplitPane, JTabbedPane, JScrollPane
b.	List Controls: JList, JComboBox
c.	Basic text components, Trees, Tables

### Books and References:

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

## Introduction to Embedded Systems and Internet of Things

### Course Objective:

- To introduce the organization and design aspects of 8051 microcontroller based embedded system.
- To make the students understand the application areas, building blocks, and characteristics of the Internet of Things.

### Course Outcome:

CO1: Describe the architecture of 8051 microcontroller and write embedded program for 8051 microcontroller.

CO2: Design the interfacing for 8051 microcontroller.

CO3: Identifying the various technologies for implementing the Internet of Things.

CO4: Understanding the prototype design and implementation of the Internet of Things.

### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSIT42</b>
<b>Course Name</b>	<b>Introduction to Embedded Systems and Internet of Things</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<b>Introduction to Embedded systems:</b> Embedded systems vs. General computing systems, Microprocessor vs. Microcontroller, Harvard vs. Von-Neumann processor/controller Architecture, RISC vs. CISC controller/Processor, Digital signal processor (DSP). <b>Sensors and Actuators:</b> Light Emitting Diode, 7-Segment LED display, Optocoupler, Stepper motor, Piezo buzzer, Keyboard, etc. <b>Memory:</b> Program storage memory (ROM), Read/Write Memory (RAM).	<b>12</b>
<b>II</b>	<b>Communication Interface:</b> On-Board communication interfaces, External communication interfaces. <b>Designing Embedded systems with 8051 microcontrollers:</b> 8-Bit Microcontroller – Architecture – Instruction Set and Programming – Programming Parallel Ports – Timers and Serial Port –Registers, SFR Registers Interrupt Handling.	<b>12</b>

<b>III</b>	<b>The Internet of Things:</b> An overview: The flavors of the Internet of Things, The “Internet” of “Things”, The Technology of the Internet of Things, The Technology of the Internet of Things. <b>Internet Principles:</b> Internet communications: An overview, IP Addresses, MAC Addresses, TCP and UDP Ports, Application layer protocols	<b>12</b>
<b>IV</b>	<b>Prototyping Embedded Devices:</b> Electronics, Embedded computer Basics, <b>Arduino:</b> Developing on the Arduino, Hardware, Openness. <b>Raspberry Pi:</b> Developing on the Raspberry Pi, Hardware, Openness.	<b>12</b>
<b>V</b>	<b>Prototyping Physical Design:</b> Preparation, Sketch, Iterate and Explore, Nondigital Methods, Laser Cutting, 3D Printing, CNC Milling. <b>Prototyping Online Components:</b> Getting started with an API, Writing a new API, Real-Time Reactions, Other protocols	<b>12</b>

### Books and References

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Introduction to embedded systems	Shibu K V	Tata Mcgraw-Hill	First	2012
2.	The 8051 Microcontroller and Embedded Systems	Muhammad Ali Mazidi	Pearson	Second	2011
3.	Embedded C	Michael J. Pont	Pearson Education	---	2007
4.	Designing the Internet of Things	Adrian McEwen, Hakim Cassimally	John Wiley and Sons	---	2014
5.	IoT (Internet of Things) Programming: A Simple and Fast Way of Learning	SharanamShah, Vaishali Shah	---	IOT Kindle Edition	---
6.	Internet of Things: A Hands-on Approach	Arshdeep Bahga, Vijay Madiseti.	VPT		2014

### Internal Evaluation: 40 Marks

20 Marks	20 Marks
Class Test	Case Study/Projects in Embedded Systems and Internet of Things

**Practical Component:**

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSITP42</b>
<b>Course Name</b>	<b>Introduction to Embedded Systems and Internet of Things Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

<b>List of Practical (using Keil/Proteus and Raspberry Pi)</b>	
1	Configure timer control registers of 8051 and develop a program to generate a given time delay
2	Port I / O: Use one of the four ports of 8051 for O/P interfaced with eight LEDs. Simulate binary counter (8 bit) on LED's
3	To demonstrate timer working in timer mode and blink LED without using any loop delay routine
4	Interface 8051 with D/A converter and generate the square wave of given frequency on the oscilloscope
5	Interface stepper motor with 8051 and write a program to move the motor through a given angle in a clockwise or counterclockwise direction.
6	Starting Raspbian OS, Familiarizing with Raspberry Pi Components and interface, Connecting to Ethernet, Monitor, and USB.
7	Displaying different LED patterns with Raspberry Pi.
8	Displaying Time over 4-Digit 7-Segment Display using Raspberry Pi
9	Controlling Raspberry Pi with WhatsApp
10	Setting up a wireless access point using Raspberry Pi.
11	Fingerprint Sensor interfacing with Raspberry Pi.
12	Raspberry Pi and GPS Module Interfacing.
13	IoT-based Web Controlled Home Automation using Raspberry Pi
14	Visitor Monitoring with Raspberry Pi and Pi Camera.
15	Building Google Assistant with Raspberry Pi.



## Computer Oriented Statistical Techniques

### Course Objective:

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

### Course Outcome:

CO1: Apply mean, median, mode, standard deviation on any given data and work with R Language.

CO2: Compare Skewness, Kurtosis, probability, sampling theory and apply statistical estimation theory and statistical decision theory

CO3: Identify the role of chi-square test for real data and apply curve fitting, method of least squares and correlation theory for any given data.

### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSIT43</b>
<b>Course Name</b>	<b>Computer Oriented Statistical Techniques</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<p><b>The Mean, Median, Mode, and Other Measures of Central Tendency:</b> 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.</p> <p><b>The Standard Deviation and Other Measures of Dispersion:</b> Dispersion, or Variation, The Range, The Mean Deviation, The SemiInterquartile Range, The 10–90 Percentile Range, The Standard Deviation, The Variance, Short Methods for Computing the Standard Deviation, Properties of the Standard Deviation, 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.</p> <p><b>Introduction to R:</b> Basic syntax, data types, variables, operators, control statements, R-functions, R – Vectors, R – lists, R Arrays.</p>	<b>12</b>

<b>II</b>	<p><b>Moments, Skewness, and Kurtosis</b> : 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.</p> <p><b>Elementary Probability Theory</b>: Definitions of Probability, Conditional Probability; Independent and Dependent Events, Mutually Exclusive Events, Mathematical Expectation, Relation Between Population, Sample Mean, and Variance, Combinations, Stirling’s Approximation to n!.</p> <p><b>Elementary Sampling Theory</b> : 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, Software Demonstration of Elementary Sampling Theory.</p>	<b>12</b>
<b>III</b>	<p><b>Statistical Estimation Theory</b>: Estimation of Parameters, Unbiased Estimates, Efficient Estimates, Point Estimates and Interval Estimates; Their Reliability, Confidence-Interval Estimates of Population Parameters, Probable Error.</p> <p><b>Statistical Decision Theory</b>: 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</p> <p><b>Statistics in R</b>: mean, median, mode, Normal Distribution , Binomial Distribution, Frequency Distribution in R.</p>	<b>12</b>
<b>IV</b>	<p><b>Small Sampling Theory</b>: Small Samples, Student’s t Distribution, Confidence Intervals, Tests of Hypotheses and Significance, The ChiSquare Distribution, Confidence Intervals for Sigma , Degrees of Freedom, The F Distribution.</p> <p><b>The Chi-Square Test</b>: 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 chisquare.</p>	<b>12</b>
<b>V</b>	<p><b>Curve Fitting and the Method of Least Squares</b>: Relationship Between Variables, Curve Fitting, Equations of Approximating Curves, Freehand Method of Curve Fitting, The Straight Line, The Method of Least Squares, Nonlinear Relationships, The Least-Squares Parabola, Regression, Applications to Time Series</p> <p><b>Correlation Theory</b>: Correlation and Regression, Linear Correlation, Measures of Correlation, The Least-Squares Regression Lines, Standard Error of Estimate, Explained and Unexplained Variation, Coefficient of Correlation,Product-Moment Formula for the Linear Correlation Coefficient.</p>	<b>12</b>

## Books and References

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Statistics	Murray r. Spiegel, Larry J. Stephens.	Mcgraw – Hill International	Fourth	
2.	A practical approach using r	R.b. Patil, H.j. Dand and R. Bhavsar	Spd	1 <sup>st</sup>	2017
3.	Fundamental of Mathematical Statistics	S.c. Gupta and v.k. Kapoor	Sultan Chand and Sons	Eleventh revised	2011
4.	Mathematical statistics	J.n. Kapur and h.c. Saxena	S. Chand	Twentieth revised	2005

## Internal Evaluation: 40 Marks

<b>20 Marks</b>	<b>20 Marks</b>
Class Test	Assignments / Problem solving

## Practical Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSITP43</b>
<b>Course Name</b>	<b>Computer Oriented Statistical Techniques Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

## List of Practical:

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

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

**Books and References:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	A Practical Approach to R Tool	R.B. Patil, H.J. Dand and R. Dahake	SPD	First	2011
2.	Statistics	Murray r. Spiegel, larry j. Stephens.	Mcgraw –hill international	Fourth	2006

## Software Engineering

### Course Objective:

The learner is introduced to the software engineering lifecycle to be applied in one or more significant application domains

### Course Outcome:

CO1: Describe various approaches like waterfall, incremental, prototyping.

CO2: Apply new software models, techniques and technologies to bring out innovative and novelistic solutions for the growth of the society in all aspects.

CO3: Develop a project by applying the software engineering principles like project management, interface design and cost estimation.

### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSIT44</b>
<b>Course Name</b>	<b>Software Engineering</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Content</b>	<b>No. of Lectures</b>
<b>I</b>	<p><b>Introduction:</b> What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc.</p> <p><b>Software Requirements:</b> Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements.</p> <p><b>Software Processes:</b> Process and Project, Component Software Processes.</p> <p><b>Software Development Process Models.</b></p> <ul style="list-style-type: none"><li>• Waterfall Model.</li><li>• Prototyping.</li><li>• Iterative Development.</li><li>• Rational Unified Process.</li><li>• The RAD Model</li><li>• Time boxing Model.</li></ul> <p><b>Agile software development:</b> Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.</p>	<b>12</b>
<b>II</b>	<p><b>Critical system:</b> Types of critical system, A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems.</p> <p><b>Requirements Engineering Processes:</b> Feasibility study, Requirements elicitation and analysis, Requirements Validations, Requirements</p>	<b>12</b>

	<p>Management.</p> <p><b>System Models:</b> Models and its types, Context Models, Behavioural Models, Data Models, Object Models, Structured Methods.</p> <p><b>Resource Allocation</b> - Introduction, Nature of Resources</p>	
III	<p><b>Architectural Design:</b> Architectural Design Decisions, System Organisation, Modular Decomposition Styles, Control Styles, Reference Architectures.</p> <p><b>User Interface Design:</b> Need of UI design, Design issues, The UI design Process, User analysis, User Interface Prototyping, Interface Evaluation.</p> <p><b>Introduction to Software Project Management:</b> Introduction, Why is Software Project Management Important? What is a Project? Project Charter, Stakeholders, The Business Case, What is Management? Management Control ,Project Portfolio Management,</p> <p><b>Quality Management:</b> Process and Product Quality, Quality assurance and Standards, Quality Planning, Quality Control, Software Measurement and Metrics.</p>	12
IV	<p><b>Verification and Validation:</b> Planning Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods.</p> <p><b>Activity Planning:</b>Introduction, Objectives of Activity Planning, Network Planning Models, Formulating a Network Model, Adding the Time Dimension, The Forward Pass, Backward Pass, Identifying the Critical Path, Activity Float, Shortening the Project Duration, Identifying Critical Activities, Activity-on-Arrow Networks.</p> <p><b>Risk Management:</b>Introduction, Risk, Categories of Risk, Risk Management Approaches, A Framework for Dealing with Risk, Risk Identification, Risk Assessment, Risk Planning, Risk Management, Evaluating Risks to the Schedule, Boehm’s Top 10 Risks and Counter Measures, Applying the PERT Technique</p> <p><b>Software Measurement:</b> Size-Oriented Metrics, Function-Oriented Metrics, Extended Function Point Metrics</p> <p><b>Software Cost Estimation:</b> Software Productivity, Estimation Techniques, Algorithmic Cost Modelling, Project Duration and Staffing</p>	12
V	<p><b>Unit Testing: Boundary Value Testing, Equivalence Class Testing, Decision Table–Based Testing, Path Testing, Data Flow Testing, Levels of Testing:</b> Introduction, Proposal Testing, Requirement Testing, Design Testing, Code Review, Unit Testing, Module Testing, Integration Testing, Big-Bang Testing, Sandwich Testing, Critical Path First, Sub System Testing, System Testing, Testing Stages.</p> <p><b>Software reuse:</b> The reuse landscape, Application frameworks, Software product lines, COTS product reuse.</p> <p><b>Distributed software engineering:</b> Distributed systems issues, Client–server computing, Architectural patterns for distributed systems, Software as a service</p>	12

## Books and References

Sr. No.	Title	Author/s	Publisher	Edition	Year
---------	-------	----------	-----------	---------	------

1.	Software Engineering, edition,	Ian Somerville	Pearson Education.	Ninth	
2.	Software Engineering	Pankaj Jalote	Narosa Publication		
3.	Software engineering, a practitioner's approach	Roger Pressman	Tata Mcgraw-hill	Seventh	
4.	Software Engineering principles and practice	WS Jawadekar	Tata Mcgraw-hill		
5.	Software Engineering A Concise Study	S.A Kelkar	PHI India.		
6.	Software Engineering Concept and Applications	Subhjit Datta	Oxford Higher Education		
7.	Software Design	D.Budgen	Pearson education	2nd	
8.	Software Engineering	KL James	PHI	EEE	2009

#### Internal Evaluation: 40 Marks

<b>20 Marks</b>	<b>20 Marks</b>
Class Test	To develop a software model using UML tools for a real time system

#### Practical Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSITP44</b>
<b>Course Name</b>	<b>Software Engineering Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

#### List of Practical: (To be executed using Star UML or any similar software)

<b>1.</b>	Study and implementation of class diagrams.
<b>2.</b>	Study and implementation of Use Case Diagrams.
<b>3.</b>	Study and implementation of Entity Relationship Diagrams.
<b>4.</b>	Study and implementation of Sequence Diagrams.
<b>5.</b>	Study and implementation of State Transition Diagrams.
<b>6.</b>	Study and implementation of Data Flow Diagrams.

<b>7.</b>	Study and implementation of Collaboration Diagrams.
<b>8.</b>	Study and implementation of Activity Diagrams.
<b>9.</b>	Study and implementation of Component Diagrams.
<b>10.</b>	Study and implementation of Deployment Diagrams.
<b>11.</b>	Advanced – DevOps project with Git, Jenkins and Docker on AWS

<b>Books and References:</b>				
<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Year</b>
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, James Rumbaugh, Ivar Jacobson	AddisonWesley	2005
4.	UML A Beginners Guide	Jason T. Roff	McGraw Hill Professional	2003



## Computer Graphics and Animation

### Course Objective:

To identify and explain the core concepts of computer graphics, apply graphics programming techniques to design and create computer graphics scenes. Students will also learn the different animation techniques

### Course Outcome:

CO1: Analyse the core concepts of graphics and working of various display devices.

CO2: Explain 2D and 3D transformation methods and construct the programs for various scan conversion, surface detection methods.

CO3: Identify the techniques used in animation and image processing.

### Theory Component:

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSIT45</b>
<b>Course Name</b>	<b>Computer Graphics and Animation</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>5</b>
<b>Credits</b>	<b>2</b>

<b>Unit</b>	<b>Contents</b>	<b>No. of Lectures</b>
<b>I</b>	<p><b>Introduction to Computer Graphics:</b>            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.</p> <p><b>Scan conversion</b> – Digital Differential Analyzer (DDA) algorithm, Bresenham's Line drawing algorithm. Bresenham's 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, Clipping Polygons, problem with multiple components.</p>	<b>12</b>
<b>II</b>	<p><b>Two Dimensional Transformation:</b> Introduction to transformations, Transformation Matrix, Types of Transformations in Two-Dimensional Graphics: Identity Transformation, Scaling, Reflection, Shear Transformations, Rotation, Translation, Rotation about an Arbitrary Point, Combined Transformation, Homogeneous Coordinates, 2D Transformations using Homogeneous Coordinates</p> <p><b>Three-dimensional transformations,</b> Objects in Homogeneous Coordinates, Three-Dimensional Transformations: Scaling,</p>	<b>12</b>

	Translation, Rotation, Shear Transformations, Reflection, World Coordinates and Viewing Coordinates, Projection, Parallel Projection, Perspective Projection.	
<b>III</b>	<b>Introduction to Solid Area Scan-Conversion</b> , Inside–Outside Test, Winding Number Method and Coherence Property, Polygon Filling, Seed Fill Algorithm, Scan-Line Algorithm, Priority Algorithm, Scan Conversion of Character, Aliasing, Anti-Aliasing, Halftoning, Thresholding and Dithering	<b>12</b>
<b>IV</b>	<b>Visible-Surface Determination:</b> 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. <b>Plane Curves and Surfaces:</b> 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.	<b>12</b>
<b>V</b>	<b>Computer Animation:</b> Principles of Animation, Key framing, Deformations, Character Animation, Physics-Based Animation, Procedural Techniques, Groups of Objects. <b>Image Manipulation and Storage:</b> 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.	<b>12</b>

### Books and References

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

**Internal Evaluation: 40 Marks**

<b>20 Marks</b>	<b>20 Marks</b>
Class Test	Develop a project using 2D or 3D methods and models.

**Practical Component:**

<b>B. Sc (Information Technology)</b>	<b>Semester – IV - SIUSITP45</b>
<b>Course Name</b>	<b>Computer Graphics and Animation Practical</b>
<b>Periods per week (1 Period is 50 minutes)</b>	<b>3</b>
<b>Credits</b>	<b>2</b>

**List of Practical:**

<b>1.</b>	Basic functions used for graphics in C
<b>2.</b>	
<b>a.</b>	Draw a co-ordinate axis at the center of the screen (name the axis as x-axis and y-axis, show the center of the screen as (320,240))
<b>b.</b>	Divide your screen into four regions, draw circle, rectangle, ellipse, and half ellipse in each region with appropriate message
<b>c.</b>	Draw a house on the screen
<b>d.</b>	Draw the following basic shapes in the center of the screen: i.Circle ii. Rectangle iii. Square iv. Concentric Circles v. Ellipse vi. Line
<b>3.</b>	
<b>a.</b>	Fill the screen with circles of same radius, with different radius (either as 1 program or 2 different programs)
<b>b.</b>	Simple text screen saver
<b>c.</b>	Display 5 Emojis for smileys
<b>d.</b>	Draw a moving car on a road
<b>4.</b>	
<b>a.</b>	Bouncing Ball
<b>b.</b>	Rocket Launching
<b>5.</b>	
<b>a.</b>	DDA Line drawing algorithm.
<b>b.</b>	Bresenham's Line drawing algorithm

<b>6.</b>	
<b>a.</b>	Mid-point line drawing algorithm
<b>b.</b>	Mid-point circle drawing algorithm
<b>7.</b>	Implement 2D scaling, translation, rotation and reflection
<b>8.</b>	Create a house and perform the following operations: i. Scaling about the origin followed by translation ii. Scaling with reference to an arbitrary point
<b>9.</b>	Implement Cohen-Sutherland line clipping algorithm
<b>10.</b>	
<b>a.</b>	Fill a rectangle using Flood Fill Algorithm
<b>b.</b>	Fill a rectangle using Boundary Fill Algorithm

### Books and References:

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
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	