# University of Mumbai

No. AAMS_UGS/ICC/2023-24/28

**CIRCULAR:-**

Attention of the Principals of the Affiliated Colleges and Directors of the Recognized Institutions in Faculty of Science & Technology is invited to this office Circular No. UG/211 of 2017-18 dated 21th August, 2017 relating to the revised syllabus as per the (CBCS) for the S.Y.B. Sc. Information Technology (Sem -III & IV).

They are hereby informed that the recommendations made by the Board of Deans at its meeting held on 27th June, 2023 **vide** item No. 6.4 (R) have been accepted by the Academic Council at its meeting held on 27th June, 2023 **vide** item No. 6.4 (R) and that in accordance therewith, the **revised syllabus of S.Y.B. Sc. (Information Technology) (CBCS) (Sem – III & IV)** has been brought into force with effect from the academic year 2023-24.

(The said circular is available on the University's website www.mu.ac.in).

(Prof. Sunil Bhirud)
I/c. REGISTRAR

MUMBAI – 400 032
13th July, 2023

To

The Principals of the Affiliated Colleges and Directors of the Recognized Institutions in Faculty of Science & Technology.

**A.C/6.4 (R) /27/06/2023**

Copy forwarded with Compliments for information to:-

1) The Dean, Faculty of Science & Technology,
2) The Chairman, Board of Studies **Information Technology**,
3) The Director, Board of Examinations and Evaluation,
4) The Director, Board of Students Development,
5) The Director, Department of Information & Communication Technology,
6) The Co-ordinator, MKCL.

**Copy for information and necessary action :-**

1. **The Deputy Registrar, College Affiliations & Development Department (CAD),**
2. **College Teachers Approval Unit (CTA),**
3. **The Deputy Registrar, (Admissions, Enrolment, Eligibility and Migration Department (AEM),**
4. **The Deputy Registrar, Academic Appointments & Quality Assurance (AAQA)**
5. **The Deputy Registrar, Research Administration & Promotion Cell (RAPC),**
6. **The Deputy Registrar, Executive Authorities Section (EA)**
   **He is requested to treat this as action taken report on the concerned resolution adopted by the Academic Council referred to the above circular.**
7. **The Deputy Registrar, PRO, Fort, (Publication Section),**
8. **The Deputy Registrar, Special Cell,**
9. **The Deputy Registrar, Fort Administration Department (FAD) Record Section,**
10. **The Deputy Registrar, Vidyanagari Administration Department (VAD),**

**Copy for information :-**

1. **The Director, Dept. of Information and Communication Technology (DICT), Vidyanagari,**
   **He is requested to upload the Circular University Website**
2. **The Director of Department of Student Development (DSD),**
3. **The Director, Institute of Distance and Open Learning (IDOL Admin), Vidyanagari,**
4. **All Deputy Registrar, Examination House,**
5. **The Deputy Registrars, Finance & Accounts Section,**
6. **The Assistant Registrar, Administrative sub-Campus Thane,**
7. **The Assistant Registrar, School of Engg. & Applied Sciences, Kalyan,**
8. **The Assistant Registrar, Ratnagiri sub-centre, Ratnagiri,**
9. **P.A to Hon'ble Vice-Chancellor,**
10. **P.A to Pro-Vice-Chancellor,**
11. **P.A to Registrar,**
12. **P.A to All Deans of all Faculties,**
13. **P.A to Finance & Account Officers, (F & A.O),**
14. **P.A to Director, Board of Examinations and Evaluation,**
15. **P.A to Director, Innovation, Incubation and Linkages,**
16. **P.A to Director, Department of Lifelong Learning and Extension (DLLE),**
17. **The Receptionist,**
18. **The Telephone Operator,**

**Copy with compliments for information to :-**

19. **The Secretary, MUASA**
20. **The Secretary, BUCTU.**

AC – 27/06/2023
Item No. 6.4 (R)

# UNIVERSITY OF MUMBAI

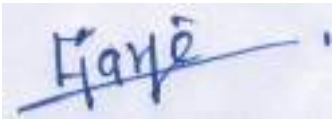
## Revised Syllabus for
## S.Y.B.Sc. (Information Technology)
## (Sem. III & IV)
## (CBCS)

(With effect from the academic year 2023-24)

# University of Mumbai

Syllabus for Approval

| Sr. No. | Heading | Particulars |
|---|---|---|
| 1 | **O: _____ Title of Course** | S.Y.B.Sc. (Information Technology) |
| 2 | **O: _____ Eligibility** | Ordinance no. O.5051<br>Circular no. UG/284 of 2007 dated 16th June 2007 |
| 3 | **R: _____ Passing Marks** | 40 % |
| 4 | No. of years/Semesters: | 3 Years/ 6 Semesters |
| 5 | Level: | ~~P.G.~~ / U.G./ ~~Diploma~~ / Certificate<br>**( Strike out which is not applicable)** |
| 6 | Pattern: | ~~Yearly~~ / Semester<br>**( Strike out which is not applicable)** |
| 7 | Status: | Revised / ~~New~~<br>**( Strike out which is not applicable)** |
| 8 | To be implemented from Academic Year : | From Academic Year: 2023-24 |



**Prof. Shivram S. Garje,**

**Dean**,

Faculty of Science and Technology

| Semester – 3 | | | |
|---|---|---|---|
| **Course Code** | **Course Type** | **Course Title** | **Credits** |
| USIT301 | Skill Enhancement Course | Python Programming | 2 |
| USIT302 | Core Subject | Data Structures | 2 |
| USIT303 | Core Subject | Computer Networks | 2 |
| USIT304 | Core Subject | Operating Systems | 2 |
| USIT305 | Core Subject | Applied Mathematics | 2 |
| USIT3P1 | Skill Enhancement Course Practical | Python Programming Practical | 2 |
| USIT3P2 | Core Subject Practical | Data Structures Practical | 2 |
| USIT3P3 | Core Subject Practical | Computer Networks Practical | 2 |
| USIT3P4 | Core Subject Practical | Operating Systems Practical | 2 |
| USIT3P5 | Core Subject Practical | Mobile Programming Practical | 2 |
| | | **Total Credits** | **20** |

| Semester – 4 | | | |
|---|---|---|---|
| **Course Code** | **Course Type** | **Course Title** | **Credits** |
| USIT401 | Skill Enhancement Course | Core Java | 2 |
| USIT402 | Core Subject | Introduction to Embedded Systems | 2 |
| USIT403 | Core Subject | Computer Oriented Statistical Techniques | 2 |
| USIT404 | Core Subject | Software Engineering | 2 |
| USIT405 | Core Subject | Computer Graphics and Animation | 2 |
| USIT4P1 | Skill Enhancement Course Practical | Core Java Practical | 2 |
| USIT4P2 | Core Subject Practical | Introduction to Embedded Systems Practical | 2 |
| USIT4P3 | Core Subject Practical | Computer Oriented Statistical Techniques Practical | 2 |
| USIT4P4 | Core Subject Practical | Software Engineering Practical | 2 |
| USIT4P5 | Core Subject Practical | Computer Graphics and Animation Practical | 2 |
| | | **Total Credits** | **20** |

# Table of Contents

# EMESTER III

# Python Programming

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Python Programming** | | **Course Code: USIT301** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objective:**

- Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.
- Express proficiency in the handling of strings and functions.
- Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.
- Identify the commonly used operations involving file systems and regular expressions.
- Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python.

| Unit | Details | Lectures |
|---|---|---|
| **I** | **Introduction:** The Python Programming Language, History, features, Installing Python, Running Python program, Debugging : Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages, The Difference Between Brackets, Braces, and Parentheses,<br>**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.<br>**Conditional Statements:** if, if-else, nested if –else<br>**Looping**: for, while, nested loops<br>**Control statements:** Terminating loops, skipping specific conditions | **12** |
| **II** | **Functions:** Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution, Parameters and Arguments, Variables and Parameters Are Local, Stack Diagrams, Fruitful Functions and Void Functions, Why Functions? Importing with from, Return Values, Incremental Development, Composition, Boolean Functions, More Recursion, Leap of Faith, Checking Types<br>**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** |

| | | |
|---|---|---|
| **III** | **Lists:** Values and Accessing Elements, Lists are mutable, traversing a List, Deleting elements from List, Built-in List Operators, Concatenation, Repetition, In Operator, Built-in List functions and methods<br>**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<br>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<br>**Files:** Text Files, The File Object Attributes, Directories<br>**Exceptions:** Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions | **12** |
| **IV** | **Regular Expressions –** Concept of regular expression, various types of regular expressions, using match function.<br>**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<br>**Multithreaded Programming:** Thread Module, creating a thread, synchronizing threads, multithreaded priority queue<br>**Modules:** Importing module, Creating and exploring modules, Math module, Random module, Time module | **12** |
| **V** | **Creating the GUI Form and Adding Widgets:**<br>**Widgets:** Button, Canvas, Checkbutton, Entry, Frame, Label, Listbox, Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel, Spinbox, PanedWindow, LabelFrame, tkMessagebox.<br>Handling Standard attributes and Properties of Widgets.<br>**Layout Management:** Designing GUI applications with proper Layout Management features.<br>**Look and Feel Customization**: Enhancing Look and Feel of GUI using different appearances of widgets.<br>**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** |

**Books and References:**

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

| | | | | | |
|---|---|---|---|---|---|
| 3. | Python GUI Programming Cookbook | Burkhard A. Meier | Packt | | 2015 |
| 4. | Introduction to Problem Solving with Python | E. Balagurusamy | TMH | 1st | 2016 |
| 5. | Murach's Python programming | Joel Murach, Michael Urban | SPD | 1st | 2017 |
| 6. | Object-oriented Programming in Python | Michael H. Goldwasser, David Letscher | Pearson Prentice Hall | 1st | 2008 |
| 7. | Exploring Python | Budd | TMH | 1st | 2016 |

**Course Outcome:**

After completing the course, the learner will be able to:

CO1: Aware of the variables, expressions, looping and conditions used in Python programming.

CO2: Implement functions, strings, lists, tuples and directories

CO3: Create GUI forms and add widgets.

CO4: Use MySQL to store data.

CO5: Apply the programming skillset learnt here into various domains by having advance programming skillset of Python and usage of libraries.

# Data Structures

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Data Structures** | | **Course Code: USIT302** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objective:**

- Ability to analyze the performance of algorithms.
- Ability to choose appropriate algorithm design techniques for solving problems.
- Understand how the choice of data structures and the algorithm design methods impact the performance of programs.

| Unit | Details | Lectures |
|---|---|---|
| **I** | **Introduction:** Data and Information, Data Structure, Classification of Data Structures, Primitive Data Types, Abstract Data Types, Data structure vs. File Organization, Operations on Data Structure, Algorithm, Importance of Algorithm Analysis, Complexity of an Algorithm, Asymptotic Analysis and Notations, Big O Notation, Big Omega Notation, Big Theta Notation, Rate of Growth and Big O Notation.<br>**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** |
| **II** | **Linked List:** Linked List, One-way Linked List, Traversal of Linked List, Searching, Memory Allocation and De-allocation, Insertion in Linked List, Deletion from Linked List, Copying a List into Other List, Merging Two Linked Lists, Splitting a List into Two Lists, Reversing One way linked List, Circular Linked List, Applications of Circular Linked List, Two way Linked List, Traversing a Two way Linked List, Searching in a Two way linked List, Insertion of an element in Two way Linked List, Deleting a node from Two way Linked List, Header Linked List, Applications of the Linked list, Representation of Polynomials, Storage of Sparse Arrays, Implementing other Data Structures. | **12** |
| **III** | **Stack**: Introduction, Operations on the Stack Memory Representation of Stack, Array Representation of Stack, Applications of Stack, Evaluation of Arithmetic Expression, Matching Parenthesis, infix and postfix operations, Recursion**.**<br>**Queue:** Introduction, Queue, Operations on the Queue, Memory Representation of Queue, Array representation of queue, Linked List Representation of Queue, Circular Queue, Some special kinds of queues, Deque, Priority Queue, Application of Priority Queue, Applications of Queues. | **12** |

| | | |
|---|---|---|
| **IV** | **Sorting and Searching Techniques**<br>Bubble, Selection, Insertion, Merge Sort.<br>Searching: Sequential, Binary, Indexed Sequential Searches.<br>**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.<br>**Advanced Tree Structures:** Red Black Tree, Operations Performed on Red Black Tree, AVL Tree, Operations performed on AVL Tree, 2-3 Tree, B-Tree. | **12** |
| **V** | **Hashing Techniques**<br>Hash function, Address calculation techniques, Common hashing functions Collision resolution, Linear probing, Quadratic, Double hashing, Bucket hashing, Deletion and rehashing<br>**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** |

**Books and References:**

| Sr. No. | Title | Author/s | Publisher | Edition | Year |
|---|---|---|---|---|---|
| 1. | A Simplified Approach to Data Structures | Lalit Goyal, Vishal Goyal, Pawan Kumar | SPD | 1st | 2014 |
| 2. | An Introduction to Data Structure with Applications | Jean – Paul Tremblay and Paul Sorenson | Tata McGraw 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 |

**Course Outcome:**

After completing the course, the learner will be able to:

CO1: Identify and distinguish data structure classification, data types, their complexities

CO2: Implement array, linked list, stack and queue.

CO3: Implement trees, various hashing techniques and graph for various applications

CO4: Compare various sorting and searching techniques

# Computer Networks

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Computer Networks** | | **Course Code: USIT303** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objective:**

- Knowledge of uses and services of Computer Network.
- Ability to identify types and topologies of network.
- Understanding of analog and digital transmission of data.
- Familiarization with the techniques of routing.
- Understand the functioning of networking application

| Unit | Details | Lectures |
|---|---|---|
| **I** | **Introduction:** Computer Network, Evolution of Computer Networks Different types of Computer Network, Difference between LAN, MAN and WAN, Hardware Devices used for Networking: Network Interface Card (NIC), Modem, Hub, Switch L1 and L2 switches, Comparison between switch and hub, Bridge, Router, Gateway.<br>Standards and administration. Network Models: Protocol layering, TCP/IP protocol suite, The OSI model. | **12** |
| **II** | **Introduction to Physical layer:** Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance.<br>**Introduction to the Data Link Layer:** Link layer addressing, Data Link Layer Design Issues, Error detection and correction, block coding<br>**Wireless LANs:** Introduction, IEEE 802.11 project, Bluetooth, WiMAX, Cellular telephony, Satellite networks. | **12** |
| **III** | **Network Layer:** IPv4 Addresses, IPv4 Protocol, ARP, ICMP, IPv6<br>**Routing:** RIP, OSPF, BGP | **12** |
| **IV** | **Transport Layer:** UDP, TCP | **12** |
| **V** | **Application Layer:** WWW, HTTP, DNS, SMTP, POP3, MIME, IMAP, DHCP, TELNET, SSH, FTP | **12** |

**Books and References:**

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

**Online Resources:**

- https://ekumbh.aicte-india.org/allbook.php
- https://free.aicte-india.org/

**Course Outcomes:**

After completing the course, the learner will be able to:

CO1: Identify various data communication standards, topologies and terminologies

CO2: Describe how signals are used to transfer data and communication aspects between nodes

CO3: Configure IP addresses using TCP/IP protocol suite

CO4: Use different application layer protocols

# Operating Systems

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Operating Systems** | | **Course Code: USIT304** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objective:**

- Analyze the concepts of processes in operating system and illustration of the scheduling of processor for a given problem instance.
- Identify the dead lock situation and provide appropriate solution so that protection and security of the operating system is also maintained.
- Analyze memory management techniques, concepts of virtual memory and disk scheduling.
- Understand the implementation of file systems and directories along with the interfacing of IO devices with the operating system.
- Ability to apply CPU scheduling algorithms to manage tasks.
- Initiation into the process of applying memory management methods and allocation policies.
- Knowledge of methods of prevention and recovery from a system deadlock.

| Unit | Details | | Lectures |
|---|---|---|---|
| **I** | **Operating System Overview:** Objectives and Functions, Evolution, Achievements, Modern Operating Systems, Fault tolerance, OS design considerations for multiprocessor and multicore, overview of different operating systems<br>**Processes:** Process Description and Control. | | **12** |
| **II** | **Threads, Concurrency:** Mutual Exclusion and Synchronization. | | **12** |
| **III** | **Concurrency:** Deadlock and Starvation,<br>**Memory:** Memory Management, Virtual Memory. | | **12** |
| **IV** | **Scheduling:** Uniprocessor Scheduling, Multiprocessor and Real-Time Scheduling | | **12** |
| **V** | **IO and File Management:** I/O Management and Disk Scheduling, File Management, **Operating System Security.** | | **12** |

**Books and References:**

| Sr. No. | Title | Author/s | Publisher | Edition | Year |
|---|---|---|---|---|---|
| 1. | Operating Systems – Internals and Design Principles | Willaim Stallings | Pearson | 9th | 2009 |
| 2. | Operating System Concepts | Abraham Silberschatz, | Wiley | 8th | |

| | | | | | |
|---|---|---|---|---|---|
| | | Peter B. Galvineg Gagne | | | |
| 3. | Operating Systems | Godbole and Kahate | McGraw Hill | 3rd | |

**Online Resources:**

- https://onlinecourses.nptel.ac.in/noc20_cs04/preview
- https://free.aicte-india.org/
- https://www.javatpoint.com/best-courses-for-the-operating-system

**Course Outcomes:**

After completing the course, the learner will be able to:

CO1: Role of Operating System Computer System.

CO2: Use the different types of Operating System and their services.

CO3: configure process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.

CO4: Apply virtual memory concepts.

CO5: Effectively use and manage secondary memory.

# Applied Mathematics

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Applied Mathematics** | | **Course Code: USIT305** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objective:**

The course is aimed to develop the basic Mathematical skills of IT students that are imperative for effective understanding of IT subjects.

- Apply the knowledge of matrices to solve the problems.
- Know and to understand various types of numerical methods.
- Ability to interpret the mathematical results in physical or practical terms for complex numbers.
- Inculcate the habit of Mathematical Thinking through Indeterminate forms and Taylor series expansion
- Solve and analyze the Partial derivatives and its application in related field of engineering

| Unit | Details | Lectures |
|---|---|---|
| **I** | **Matrices:** Inverse of a matrix, Properties of matrices, Elementary Transformation, Rank of Matrix, Echelon or Normal Matrix, Inverse of matrix, Linear equations, Linear dependence and linear independence of vectors, Linear transformation, Characteristics roots and characteristics vectors, Properties of characteristic vectors, Caley-Hamilton Theorem, Similarity of matrices, Reduction of matrix to a diagonal matrix which has elements as characteristics values.<br>**Complex Numbers:** Complex number, Equality of complex numbers, Graphical representation of complex number(Argand's Diagram), Polar form of complex numbers, Polar form of x+iy for different signs of x,y, Exponential form of complex numbers, Mathematical operation with complex numbers and their representation on Argand's Diagram, Circular functions of complex angles, Definition of hyperbolic function, Relations between circular and hyperbolic functions, Inverse hyperbolic functions, Differentiation and Integration, Graphs of the hyperbolic functions, Logarithms of complex quality, j(=i)as an operator(Electrical circuits) | **12** |
| **II** | **Equation of the first order and of the first degree:** Separation of variables, Equations homogeneous in x and y, Non-homogeneous linear equations, Exact differential Equation, Integrating Factor, Linear Equation and equation reducible to this form, Method of substitution.<br>**Differential equation of the first order of a degree higher than the first:** Introduction, Solvable for p (or the method of factors), Solve for | **12** |

| | | |
|---|---|---|
| | y, Solve for x, Clairaut's form of the equation, Methods of Substitution, Method of Substitution.<br>**Linear Differential Equations with Constant Coefficients:** Introduction, The Differential Operator, Linear Differential Equation f(D) y = 0, Different cases depending on the nature of the root of the equation f(D) = 0, Linear differential equation f(D) y = X, The 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. | |
| **III** | **The Laplace Transform:** Introduction, Definition of the Laplace Transform, Table of Elementary Laplace Transforms, Theorems on Important Properties of Laplace Transformation, First Shifting Theorem, Second Shifting Theorem, The Convolution Theorem, Laplace Transform of an Integral, Laplace Transform of Derivatives, **Inverse Laplace Transform:** Shifting Theorem, Partial fraction Methods, Use of Convolution Theorem, Solution of Ordinary Linear Differential Equations with Constant Coefficients, Solution of Simultaneous Ordinary Differential Equations, Laplace Transformation of Special Function, Periodic Functions, Heaviside Unit Step Function, Dirac-delta Function(Unit Impulse Function), | **12** |
| **IV** | **Multiple Integrals:** Double Integral, Change of the order of the integration, Double integral in polar co-ordinates, Triple integrals.<br>**Applications of integration:** Areas, Volumes of solids. | **12** |
| **V** | **Beta and Gamma Functions –** Definitions**,** Properties and Problems. Duplication formula.<br>**Differentiation Under the Integral Sign**<br>**Error Functions** | **12** |

**Books and References:**

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

**Course Outcomes:**

Upon the successful completion of the course, students will be able to:

**CO 1:** Solve the matrix operations, identify the linear dependence and independence of a vectors.

**CO 2:** Familiar with the various forms and operations of a complex number.
**CO 3:** Find the Laplace transform of a function and Inverse Laplace transform of a function using definition also solve ordinary differential equations using Laplace transform.
**CO 4:** Evaluate the multiple integrals in Cartesian, Polar coordinates, change the order of the integral,
**CO 5:** Apply integration methods to calculate the areas and volumes of solids.
**CO 6:** Evaluate the Beta, Gamma, Differentiation Under integral sign and error functions

# Python Programming Practical

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Python Programming Practical** | | **Course Code: USIT3P1** | |
| **Periods per week (1 Period is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | **Internal** | **--** | **--** |

| **List of Practical** | |
|---|---|
| **1.** | **Write the program for the following:** |
| a. | Create a program that asks the user to enter their name and their age. Print out a message addressed to them that tells them the year that they will turn 100 years old. |
| b. | Enter the number from the user and depending on whether the number is even or odd, print out an appropriate message to the user. |
| c. | Write a program to generate the Fibonacci series. |
| d. | Write a function that reverses the user defined value. |
| e. | Write a function to check the input value is Armstrong and also write the function for Palindrome. |
| f. | Write a recursive function to print the factorial for a given number. |
| | |
| **2.** | **Write the program for the following:** |
| a. | Write a function that takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise. |
| b. | Define a function that computes the *length* of a given list or string. |
| c. | Define a *procedure* histogram() that takes a list of integers and prints a histogram to the screen. For example, histogram([4, 9, 7]) should print the following:<br><br>****<br>*********<br>******* |
| | |
| **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:<br><br>a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]<br><br>and write a program that prints out all the elements of the list that are less than 5. |
| | |
| **4.** | **Write the program for the following:** |

| | |
|---|---|
| a. | Write a program that takes two lists and returns True if they have at least one common member. |
| b. | Write a Python program to print a specified list after removing the 0th, 2nd, 4th and 5th elements. |
| c. | Write a Python program to clone or copy a list |
| | |
| **5.** | **Write the program for the following:** |
| a. | Write a Python script to sort (ascending and descending) a dictionary by value. |
| b. | Write a Python script to concatenate following dictionaries to create a new one.<br>Sample Dictionary :<br>dic1={1:10, 2:20}<br>dic2={3:30, 4:40}<br>dic3={5:50,6:60}<br>Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60} |
| c. | Write a Python program to sum all the items in a dictionary. |
| | |
| **6.** | **Write the program for the following:** |
| a. | Write a Python program to read an entire text file. |
| b. | Write a Python program to append text to a file and display the text. |
| c. | Write a Python program to read last n lines of a file. |
| | |
| **7.** | **Write the program for the following:** |
| a. | Design a class that store the information of student and display the same |
| b. | Implement the concept of inheritance using python |
| c. | Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers).<br>i. Write a method called add which returns the sum of the attributes x and y.<br>ii. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER.<br>iii. Write a static method called subtract, which takes two number parameters, b and c, and returns b - c.<br>iv. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and a deleter for manipulating the values of x and y. |
| | |
| **8.** | **Write the program for the following:** |
| a. | Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the "Control Flow and Functions" exercise into this file and save it.<br>Now open a new file and save it in the same directory. You should now be able to import your own module like this:<br>import geometry |

| | |
|---|---|
| | Try and add print dir(geometry) to the file and run it.<br><br>Now write a function pointyShapeVolume(x, y, squareBase) that calculates the volume of a square pyramid if squareBase is True and of a right circular cone if squareBase is False. x is the length of an edge on a square if squareBase is True and the radius of a circle when squareBase is False. y is the height of the object. First use squareBase to distinguish the cases. Use the circleArea and squareArea from the geometry module to calculate the base areas. |
| b. | Write a program to implement exception handling. |
| | |
| **9.** | **Write the program for the following:** |
| a. | Try to configure the widget with various options like: bg=”red”, family=”times”, size=18 |
| b. | Try to change the widget type and configuration options to experiment with other widget types like Message, Button, Entry, Checkbutton, Radiobutton, Scale etc. |
| | |
| **10.** | **Design the database applications for the following:** |
| a. | Design a simple database application that stores the records and retrieve the same. |
| b. | Design a database application to search the specified record from the database. |
| c. | Design a database application to that allows the user to add, delete and modify the records. |
| | |

**Books and References:**

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

# Data Structures Practical

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Data Structures Practical** | | **Course Code: USIT3P2** | |
| **Periods per week (1 Period is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | **Internal** | **--** | **--** |

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

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

**Books and References:**

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

# Computer Network Practical

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Computer Network Practical** | | **Course Code: USIT3P3** | |
| **Periods per week (1 Period is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | **Internal** | **--** | |

| **List of Practical:** | |
|---|---|
| **1.** | **Colour code for crimping LAN (Cat 5/6/7) cable** |
| a. | Study of Different color codes |
| b. | Study of different connecting devices and their differences |
| c. | Crimping LAN Cable |
| | |
| **2.** | **Configuring LAN setup** |
| a. | Planning and Setting IP networks |
| b. | Configuring subnet |
| c. | Study of basic network command and Network configuration commands. ipconfig, netstat, ARP, ping, trace route etc. |
| d. | Basic network troubleshooting. |
| e. | Configuration of TCP/IP Protocols in Windows / Linux. |
| f. | Implementation of Drive/file sharing and printer sharing. |
| | |
| **3.** | **IPv4 Addressing and Subnetting** |
| a. | Given an IP address and network mask, determine other information about the IP address such as:<br>• Network address<br>• Network broadcast address<br>• Total number of host bits<br>• Number of hosts |
| b. | Given an IP address and network mask, determine other information about the IP address such as:<br>• The subnet address of this subnet<br>• The broadcast address of this subnet<br>• The range of host addresses for this subnet<br>• The maximum number of subnets for this subnet mask<br>• The number of hosts for each subnet<br>• The number of subnet bits<br>• The number of this subnet |
| | |
| **4.** | **Designing and configuring a network topology** |
| a. | Configure IP static routing |

| | |
|---|---|
| | |
| **5.** | Configure IP routing using RIP. |
| | |
| **6.** | Configuring Simple and multi-area OSPF. |
| | |
| **7.** | **Configuring server and client.** |
| **a.** | Configure DHCP |
| **b.** | Configure DNS |
| **c.** | Configure HTTP |
| **d.** | Configure Telnet |
| **e.** | Configure FTP |
| | |
| **8.** | Configure basic security features for networks |
| | |
| **9.** | Packet capture and header analysis by wire-shark (TCP, UDP, IP etc.) |
| | |
| **10.** | **Planning and Design a corporate network for a given scenario.** |
| | |

# Operating System Practical

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Operating System Practical** | | **Course Code: USIT3P4** | |
| **Periods per week (1 Period is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | **Internal** | **--** | |

| **List of Practical:** | |
|---|---|
| **1.** | **Installation and Configuration of virtual machine** |
| a. | Installation of virtual machine software. |
| b. | Installation of Windows OS |
| c. | Installation of Linux OS |
| | |
| **2.** | **Windows (DOS) Commands** |
| a. | Date, time, prompt, md, cd, rd, path. |
| b. | Chkdsk, copy, xcopy, format, fidsk, cls, defrag, del, move. |
| c. | Diskcomp, diskcopy, diskpart, doskey, echo |
| d. | Edit, fc, find, rename, set, type, ver |
| | |
| **3.** | **Linux commands:** |
| a. | pwd, cd, absolute and relative paths, ls, mkdir, rmdir |
| b. | file, touch, rm, cp. mv, rename, head, tail, cat, tac, more, less, strings, chmod |
| c. | ps, top, kill, pkill, bg, fg |
| d. | grep, locate, find, locate |
| e. | date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which |
| f. | Compression: tar, gzip |
| | |
| **4.** | **Working with Linux Desktop and utilities** |
| a. | The vi editor |
| b. | Graphics User Interface |
| c. | Working with Terminal |
| d. | Adjusting display resolution |
| e. | Using the browsers |
| f. | Configuring simple networking |
| g. | Creating users and shares |
| | |
| **5.** | **Installing utility software on Linux and Windows** |
| | |
| **6.** | **Running C/C++/Python programs in Linux** |
| | |
| **7.** | **Introduction to Linux Shell Scripting** |

| | |
|---|---|
| **a.** | Basic operators |
| **b.** | Decision Making |
| **c.** | Looping |
| **d.** | Regular Expression |
| **e.** | Special variables and command Line arguments |
| | |
| **8.** | **Case study of Server OS:** Windows Server 2022 operating System - Architecture, Components, Services, Configuration |
| | |
| **9.** | **Case study of Android OS:** Architecture, Components, Services, Configuration |
| | |
| **10.** | **Case study of Cloud OS:** AWS, Azure, Google Cloud |

# Mobile Programming Practical

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Mobile Programming Practical** | | **Course Code: USIT3P5** | |
| **Periods per week (1 Period is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | **Internal** | **--** | **--** |

**The practical's will be based on HTML5, CSS, Flutter. (Android will be introduced later after they learn Java)**

| **List of Practical** | |
|---|---|
| | Setting up Flutter, PhoneGAP Project and environment. |
| 1. | Program to demonstrate the features of Dart language. |
| | |
| 2. | Designing the mobile app to implement different widgets. |
| | |
| 3. | Designing the mobile app to implement different Layouts. |
| | |
| 4. | Designing the mobile app to implement Gestures. |
| | |
| 5. | Designing the mobile app to implement the theming and styling. |
| | |
| 6. | Designing the mobile app to implement the routing. |
| | |
| 7. | Designing the mobile app to implement the animation. |
| | |
| 8. | Designing the mobile app to implement the state management. |
| | |
| 9. | Designing the mobile app working with SQLite Database. |
| | |
| 10. | Designing the mobile app working with Firebase. |

| **Books and References:** | | | | | |
|---|---|---|---|---|---|
| **Sr. No.** | **Title** | **Author/s** | **Publisher** | **Edition** | **Year** |
| 1. | Flutter for Beginners | Alessandro Biessek | Packt Publishing | | 2019 |
| 2. | PhoneGap By Example | Andrey Kovalenko | PACKT Publishing | 1st | 2015 |

# SEMESTER IV

# Java Programming

| B. Sc. (Information Technology) | | Semester – IV | |
|---|---|---|---|
| **Course Name: Java Programming** | | **Course Code: USIT401** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objectives:**

Upon completion of this course, students will be able to:

- Understand the concept of OOP as well as the purpose and usage principles of inheritance, polymorphism, encapsulation and method overloading.
- Identify classes, objects, members of a class and the relationships among them needed for a specific problem.
- Create Java application programs using sound OOP practices (e.g., interfaces and APIs) and proper program structuring (e.g., by using access control identifies, automatic documentation through comments, error exception handling).
- Use testing and debugging tools to automatically discover errors of Java programs as well as use versioning tools for collaborative programming/editing.
- Develop programs using the Java Collection API as well as the Java standard class library.
- Apply object-oriented programming concepts in problem solving through JAVA.

| Unit | Details | Lectures |
|---|---|---|
| **I** | **Introduction:** History, Features of Java, Java Development Kit, Java Application Programming Interface, Java Virtual Machine, Java Program Structure.<br>**Classes:** The Class Object and Its Attributes, Class Methods, Accessing A Method, Method Overloading, Instantiating Objects from A Class, Constructors, this keyword, super keyword, Types of Classes, Scope Rules, Access Modifier, constants, static members of a class, garbage collection. | **12** |
| **II** | **Inheritance:** Derived Class Objects, Inheritance and Access Control, Default Base Class Constructors, this and super keywords. Abstract Classes and Interfaces, Abstract Classes, Abstract Methods,<br>**Interfaces**: What Is an Interface? How Is an Interface Different from An Abstract Class? Multiple Inheritance, Defining an Interface, Implementing Interfaces. | **12** |
| **III** | **Exceptions:** Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause, Built-in Exceptions in java<br>**Multithreading:** Thread Creations, Thread Life Cycle, Life Cycle Methods, Synchronization, wait() notify() notify all() methods<br>**Packages:** Introduction to predefined packages, User Defined Packages, Access specifier, Java Built-in packages, Array Class, String Class | **12** |

| | | |
|---|---|---|
| **IV** | **Introduction to JFC and Swing-** Features of the Java Foundation Classes, Swing API Components, JComponent Class, Containers and Panels, Labels, Buttons, RadioButton, Check Boxes, Text-Entry Components, Menus<br>**Layouts:** Flow Layout, Grid Layout, Border Layout<br>**Event Handling:** Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes. | **12** |
| **V** | **Advanced Swing Controls:** JScrollPane, Lists and Combo Boxes, Colors and File Choosers, Tables and Trees, JTabbedPane**.**<br>**JDBC**: Introduction, JDBC Architecture, JDBC Drivers, java.sql package, Using Statement, PreparedStatement, CallableStatement, ResultSet | **12** |

**Books and References:**

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

**Course Outcome:**

After completing the course, the learner will be able to:

CO1: Learn the architecture of Java

CO2: Identify data types, control flow, classes, inheritance, exceptions and event handling

CO3: Use object-oriented concepts for problem solving real-life applications

CO4: Build GUI programs

CO5 : Create event driven programs using java.

# Introduction to Embedded Systems

| B. Sc. (Information Technology) | | Semester – IV | |
|---|---|---|---|
| **Course Name: Introduction to Embedded Systems** | | **Course Code: USIT402** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objectives:**

- To introduce the Building Blocks of Embedded System
- To Educate in Various microcontrollers used in Embedded Development
- To Introduce Bus Communication in processors, Input/output interfacing.
- To impart knowledge in sensors and actuators.
- To familiar with the real world application development using embedded system.

| Unit | Details | Lectures |
|---|---|---|
| **I** | **PIC MICROCONTROLLER:** Architecture – memory organization – addressing modes – instruction set – PIC programming in Assembly & C –I/O port, Data Conversion, RAM & ROM Allocation, Timer programming<br>**Advanced ARM Controllers:** Introduction to ARM and its Features, Architecture – memory organization – addressing modes –The ARM Programmer's model -Registers – Pipeline - Interrupts – Coprocessors – Interrupt Structure | **12** |
| **II** | **Communication Protocol & Implementation:** Introduction to Communication Protocol, I2C - Interfacing with micro controller using bit-banking method, I2C devices – RTC, Memory, ADC-DAC, Port Expander, SPI (Serial Peripheral Interface), Bluetooth, Wi-Fi and RFID. Understanding Serial, Communication, Bluetooth Communication, SPI Interface ZigBee, Wi-Fi, $I^2C$, Infrared, RFID, GSM, GPS, PDH/SDH/Ethernet | **12** |
| **III** | **Getting Started with Arduino:** Introduction, Arduino Variants, Install the Drivers, Arduino IDE<br>**Basic Functions:** Overview, Structure, Digital I/O Functions, Analog I/O Functions, Advanced I/O Functions, Timer Functions, Communication Functions, Interrupt Functions, Math Functions, Programming Language Reference | **12** |

| | | |
|---|---|---|
| **IV** | **Using Sensors with the Arduino:** Light Sensitive Sensors, Temperature Sensors, Temperature and Humidity Sensor, Line-Tracking Sensor, Ultrasonic Sensors, Digital Infrared Motion Sensor, Joystick Module, Gas Sensor, Hall Sensor, Color Sensor, Digital Tilt Sensor, Triple Axis Acceleration Sensor, Analog Sound Sensor, Voice Recognition Module, Digital Vibration Sensor, Flame Sensor, Capacitive Touch Sensor<br>**Electromechanical Control Using the Arduino:** DC Motor, Stepper Motor, Servo Motor | **12** |
| **V** | **Wireless Control Using the Arduino:** Infrared Transmitter and Receiver, Wireless Radio Frequency, Bluetooth, GSM/GPRS, Wi-Fi<br>**Case Studies:**<br>• Air Quality Monitor Using Arduino<br>• A Fire-Fighting Robot Using Arduino<br>• Intelligent Lock System Using Arduino | **12** |

**Books and References:**

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

**Course Outcome:**

CO1: Differentiate between general purpose and embedded systems

CO2: Discuss the characteristics and quality attributes of embedded systems

CO3: Use different types of sensors for appropriately

CO4: Design and develop embedded systems

# Computer Oriented Statistical Techniques

| B. Sc. (Information Technology) | | Semester – IV | |
|---|---|---|---|
| **Course Name: Computer Oriented Statistical Techniques** | | **Course Code: USIT403** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objectives:**

1. To learn the different methods of calculating the central tendencies.
2. To introduce the moments, skewness and kurtosis.
3. To learn scientific view to conduct the survey in proper way to collect the data about specific perspective.
4. To Learn variety of probability sampling methods for selecting a sample from a population.
5. To learn the sampling theory and testing of hypothesis and making inferences.
6. To introduce the students with understanding of the curve fitting, regression and correlation techniques.

| Unit | Details | Lectures |
|---|---|---|
| **I** | **The Mean, Median, Mode, and Other Measures of Central Tendency**: Index, or Subscript, Notation, Summation Notation, Averages, or Measures of Central Tendency ,The Arithmetic Mean , The Weighted Arithmetic Mean ,Properties of the Arithmetic Mean ,The Arithmetic Mean Computed from Grouped Data ,The Median ,The Mode, The Empirical Relation Between the Mean, Median, and Mode, The Geometric Mean G, The Harmonic Mean H ,The Relation Between the Arithmetic, Geometric, and Harmonic Means, The Root Mean Square, Quartiles, Deciles, and Percentiles, Software and Measures of Central Tendency.<br>**The Standard Deviation and Other Measures of Dispersion**: Dispersion, or Variation, The Range, The Mean Deviation, The Semi-Interquartile Range, The 10–90 Percentile Range, The Standard Deviation, The Variance, Short Methods for Computing the Standard Deviation, Properties of the Standard Deviation, Charlie's Check, Sheppard's Correction for Variance, Empirical Relations Between Measures of Dispersion, Absolute and Relative Dispersion; Coefficient of Variation, Standardized Variable; Standard Scores, Software and Measures of Dispersion.<br>**Introduction to R:** Basic syntax, data types, variables, operators, control statements, R-functions, R –Vectors, R – lists, R Arrays. | **12** |

| | | |
|---|---|---|
| **II** | **Moments, Skewness, and Kurtosis** : Moments , Moments for Grouped Data ,Relations Between Moments , Computation of Moments for Grouped Data, Charlie's Check and Sheppard's Corrections, Moments in Dimensionless Form, Skewness, Kurtosis, Population Moments, Skewness, and Kurtosis, Software Computation of Skewness and Kurtosis.<br>**Elementary Probability Theory**: Definitions of Probability, Conditional Probability; Independent and Dependent Events, Mutually Exclusive Events, Probability Distributions, Mathematical Expectation, Relation Between Population, Sample Mean, and Variance, Combinatorial Analysis, Combinations, Stirling's Approximation to n!, Relation of Probability to Point Set Theory, Euler or Venn Diagrams and Probability.<br>**Elementary Sampling Theory** : Sampling Theory, Random Samples and Random Numbers, Sampling With and Without Replacement, Sampling Distributions, Sampling Distribution of Means, Sampling Distribution of Proportions, Sampling Distributions of Differences and Sums, Standard Errors, Software Demonstration of Elementary Sampling Theory. | **12** |
| **III** | **Statistical Estimation Theory**: Estimation of Parameters, Unbiased Estimates, Efficient Estimates, Point Estimates and Interval Estimates; Their Reliability, Confidence-Interval Estimates of Population Parameters, Probable Error.<br>**Statistical Decision Theory:** Statistical Decisions, Statistical Hypotheses, Tests of Hypotheses and Significance, or Decision Rules, Type I and Type II Errors, Level of Significance, Tests Involving Normal Distributions, Two-Tailed and One-Tailed Tests, Special Tests, Operating-Characteristic Curves; the Power of a Test, p-Values for Hypotheses Tests, Control Charts, Tests Involving Sample Differences, Tests Involving Binomial Distributions.<br>**Statistics in R:** mean, median, mode, Normal Distribution , Binomial Distribution, Frequency Distribution in R. | **12** |
| **IV** | **Small Sampling Theory**: Small Samples, Student's t Distribution, Confidence Intervals, Tests of Hypotheses and Significance, The Chi-Square Distribution, Confidence Intervals for Sigma , Degrees of Freedom, The F Distribution.<br>**The Chi-Square Test**: Observed and Theoretical Frequencies, Definition of chi-square, Significance Tests, The Chi-Square Test for Goodness of Fit, Contingency Tables, Yates' Correction for Continuity, Simple Formulas for Computing chi-square, Coefficient of Contingency, Correlation of Attributes, Additive Property of chi-square. | **12** |
| **V** | **Curve Fitting and the Method of Least Squares:** Relationship Between Variables, Curve Fitting, Equations of Approximating Curves, Freehand Method of Curve Fitting, The Straight Line, The Method of Least Squares, The Least-Squares Line, Nonlinear Relationships, The | **12** |

| | | |
|---|---|---|
| | Least-Squares Parabola, Regression, Applications to Time Series, Problems Involving More Than Two Variables.<br>**Correlation Theory:** Correlation and Regression, Linear Correlation, Measures of Correlation, The Least-Squares Regression Lines, Standard Error of Estimate, Explained and Unexplained Variation, Coefficient of Correlation, Remarks Concerning the Correlation Coefficient, Product-Moment Formula for the Linear Correlation Coefficient, Short Computational Formulas, Regression Lines and the Linear Correlation Coefficient, Correlation of Time Series, Correlation of Attributes, Sampling Theory of Correlation, Sampling Theory of Regression. | |

**Books and References:**

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

**Course Outcome: Upon the successful completion of the course, students will be able to:**

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

# Software Engineering

| B. Sc. (Information Technology) | | Semester – IV | |
|---|---|---|---|
| **Course Name: Software Engineering** | | **Course Code: USIT404** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objective:**

- Develop the software projects or prototypes by understanding the requirements.
- Meet the project deadlines along with the number of resources and type of tasks to be carried out.
- Evaluate and analyze the SDLC and basic architecture SRS documents.
- Help to understand the software design and coding techniques.
- Understand the software testing principles.
- Understand the concept project management.
- Identify various concepts of Advanced UML techniques

| Unit | Details | Lectures |
|---|---|---|
| **I** | **Introduction:** What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc.<br>**Software Requirements:** Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements.<br>**Software Processes:**<br>Process and Project, Component Software Processes.<br>**Software Development Process Models.**<br>• Waterfall Model.<br>• Prototyping.<br>• Iterative Development.<br>• Rational Unified Process.<br>• The RAD Model<br>• Time boxing Model.<br>**Agile software development**: Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods. | **12** |
| **II** | **Socio-technical system:** Essential characteristics of socio technical systems, Emergent System Properties, Systems Engineering, Components of system such as organization, people and computers, Dealing Legacy Systems.<br>**Critical system:** Types of critical system, A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems. | **12** |

| | | |
|---|---|---|
| | **Requirements Engineering Processes:** Feasibility study, Requirements elicitation and analysis, Requirements Validations, Requirements Management.<br>**System Models:** Models and its types, Context Models, Behavioural Models, Data Models, Object Models, Structured Methods. | |
| **III** | **Architectural Design:** Architectural Design Decisions, System Organisation, Modular Decomposition Styles, Control Styles, Reference Architectures.<br>**User Interface Design:** Need of UI design, Design issues, The UI design Process, User analysis, User Interface Prototyping, Interface Evaluation.<br>**Project Management**<br>Software Project Management, Management activities, Project Planning, Project Scheduling, Risk Management.<br>**Quality Management:** Process and Product Quality, Quality assurance and Standards, Quality Planning, Quality Control, Software Measurement and Metrics. | **12** |
| **IV** | **Verification and Validation:** Planning Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods. **Software Testing:** System Testing, Component Testing, Test Case Design, Test Automation.<br>**Software Measurement**: Size-Oriented Metrics, Function-Oriented Metrics, Extended Function Point Metrics<br>**Software Cost Estimation:** Software Productivity, Estimation Techniques, Algorithmic Cost Modelling, Project Duration and Staffing | **12** |
| **V** | **Process Improvement:** Process and product quality, Process Classification, Process Measurement, Process Analysis and Modeling, Process Change, The CMMI Process Improvement Framework.<br>**Service Oriented Software Engineering:** Services as reusable components,<br>Service Engineering, Software Development with Services.<br>**Software reuse:** The reuse landscape, Application frameworks, Software product lines, COTS product reuse.<br>**Distributed software engineering**: Distributed systems issues, Client–server computing, Architectural patterns for distributed systems, Software as a service | **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 | Subhajit Datta | Oxford Higher Education | | |
| 7. | Software Design | D.Budgen | Pearson education | 2nd | |
| 8. | Software Engineering | KL James | PHI | EEE | 2009 |

**Course Outcome:**

After completing the course, the learner will be able to:

CO1: Understand software engineering

CO2: Apply software engineering principles

CO3: Discuss various approaches to verification and validation of software including testing, measurements and estimation of software products

CO4: Create software using different software development models

# Computer Graphics and Animation

| B. Sc. (Information Technology) | | Semester – IV | |
|---|---|---|---|
| **Course Name: Computer Graphics and Animation** | | **Course Code: USIT405** | |
| **Periods per week (1 Period is 50 minutes)** | | **5** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2½** | **75** |
| | **Internal** | **--** | **25** |

**Course Objectives:**

1. To train the students to acquire skills in generating marketable computer graphics and animated pictures, especially in the area of advertisements.
2. To train the students to acquire skills and mastery in the use of different software producing graphics and animation.
3. The course introduces the basic concepts of computer graphics.
4. It provides the necessary theoretical background and demonstrates the application of computer science to graphics.
5. The course further allows students to develop programming skills in computer graphics through programming assignments.

| Unit | Details | Lectures |
|---|---|---|
| **I** | **Introduction to Computer Graphics:**<br>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.<br>**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** |
| **II** | **Two-Dimensional Transformations:**<br>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 | **12** |

| | | |
|---|---|---|
| | through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations.<br>**Three-Dimensional Transformations:**<br>Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-Dimensional Translation, Multiple Transformation, Rotation about an Arbitrary Axis in Space, Reflection through an Arbitrary Plane, Matrix Representation of 3D Transformations, Composition of 3D Transformations, Affine and Perspective Geometry, Perspective Transformations, Techniques for Generating Perspective Views, Vanishing Points, the Perspective Geometry and camera models, Orthographic Projections, Axonometric Projections, Oblique Projections, View volumes for projections. | |
| **III** | **Viewing in 3D**<br>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.<br>**Light:** Radiometry, Transport, Equation, Photometry<br>**Color:** Colorimetry, Color Spaces, Chromatic Adaptation, Color Appearance | **12** |
| **IV** | **Visible-Surface Determination:**<br>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.<br>**Plane Curves and Surfaces:**<br>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** |
| **V** | **Computer Animation:**<br>Principles of Animation, Key framing, Deformations, Character Animation, Physics-Based Animation, Procedural Techniques, Groups of Objects.<br>**Image Manipulation and Storage:**<br>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. | **12** |

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

**After completion of the course students are supposed to be able to:**

**CO 1.** Understand the basics of computer graphics, different graphics systems and applications of computer graphics
**CO 2.** Compare various algorithms for scan conversion and filling of basic objects
**CO 3.** Use of geometric transformations on graphics objects and their application in composite form.
**CO 4.** Extract scene with different clipping methods and its transformation to graphics display device.
**CO 5.** Explore projections and visible surface detection techniques for display of 3D scene on 2D screen.
**CO 6.** Render projected objects to naturalize the scene in 2D view and use of illumination models
**CO 7.** Understand the core concepts and mathematical foundations of computer graphics
**CO 8.** Know the fundamental computer graphics algorithms and data structures
**CO 9.** Understand an overview of different modeling approaches and methods
**CO 10.** Apply basic shading and texture mapping techniques
**CO 11.** Understand light interaction with 3D scenes
**CO 12.** Explain the applications, areas, and graphic pipeline, display and hardcopy technologies.
**CO 13.** Apply and compare the algorithms for drawing 2D images also explain aliasing, anti-aliasing and half toning techniques.
**CO 14.** Discuss OpenGL application programming Interface and apply it for 2D & 3D computer graphics.
**CO 15.** Analyze and apply clipping algorithms and transformation on 2D images.
**CO 16.** Solve the problems on viewing transformations and explain the projection and hidden surface removal algorithms.
**CO 17.** Apply basic ray tracing algorithm, shading, shadows, curves and surfaces and also solve the problems of curves.

# Java Programming Practical

| B. Sc. (Information Technology) | | Semester – III | |
|---|---|---|---|
| **Course Name: Java Programming Practical** | | **Course Code: USIT4P----** | |
| **Periods per week (1 Period is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | **Internal** | **--** | |

| **List of Practical:** | |
|---|---|
| **1.** | **OOPs concepts in Java – 1** |
| a. | Write a program to create a class and implement a default, overloaded and copy Constructor. |
| b. | Write a program to create a class and implement the concepts of Method Overloading |
| c. | Write a program to create a class and implement the concepts of Static methods |
| | |
| **2.** | **OOPs concepts in Java – 2** |
| a. | Write a program to implement the concepts of Inheritance and Method overriding |
| b. | Write a program to implement the concepts of Abstract classes and methods |
| c. | Write a program to implement the concept of interfaces |
| | |
| **3.** | **Exceptions** |
| a. | Write a program to raise built-in exceptions and raise them as per the requirements |
| b. | Write a program to define user defined exceptions and raise them as per the requirements |
| | |
| **4.** | **Multithreading:** Write a java application to demonstrate 5 bouncing balls of different colors using threads. |
| | |
| **5.** | **JDBC** |
| a. | Write a JDBC program that displays the data of a given table in a GUI Table. |
| b. | Write a JDBC program to Show the details of a specified product from a given table selected using Combobox. |
| c. | Write a GUI application to Navigate forward and reverse result set data. |
| | |
| **6.** | **Swing** |
| a. | Create a swing application that randomly changes color on button click. |
| b. | Create a Swing application to demonstrate use of TextArea using scrollpane to show contest of text file in textarea selected using file chooser. |
| c. | Create a Swing application to demonstrate use of scrollpane to change its color selected using colour chooser. |
| | |
| **7.** | **Layouts: Write programs for the following layouts:** |

| | |
|---|---|
| **a.** | Flow Layout |
| **b.** | Grid Layout |
| **c.** | Border Layout |
| | |
| **8.** | **Events: Write programs to demonstrate the following events:** |
| **a.** | ActionEvent |
| **b.** | MouseEvent |
| **c.** | KeyEvent |
| **d.** | SelectionEvent |
| **e.** | FocusEvent |
| | |
| **9.** | **Demonstrate the use of Adapter Class in Event Handling** |
| | |
| **10.** | **Demonstrate the use of Anonymous Inner Class in Event Handling** |

**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 Practical

| B. Sc. (Information Technology) | | Semester – IV | |
|---|---|---|---|
| **Course Name: Introduction to Embedded Systems Practical** | | **Course Code: USIT4P2** | |
| **Periods per week**<br>**1 Period is 50 minutes** | **Lectures per week** | **3** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | | | |

| **List of Practical: All practicals to be done online using TinkerCAD** | |
|---|---|
| **1.** | **Introduction to Arduino** |
| | Introduction to Arduino circuits and breadboarding |
| | Blinking of LEDs |
| | |
| **2.** | Program using Light Sensitive Sensors |
| | |
| **3.** | Program using temperature sensors |
| | |
| **4.** | Programs using humidity sensors |
| | |
| **5.** | Programs using Line tracking sensors |
| | |
| **6.** | Programs using Ultrasonic Sensors |
| | |
| **7.** | Programs using digital infrared motion sensors |
| | |
| **8.** | Programs using gas sensors |
| | |
| **9.** | Programs using servo motors |
| | |
| **10.** | Programs making Joystick with Arduino |

## Computer Oriented Statistical Techniques Practical

| **B. Sc. (Information Technology)** | | **Semester – IV** | |
|---|---|---|---|
| **Course Name: Computer Oriented Statistical Techniques Practical** | | **Course Code: USIT4P3** | |
| **Periods per week**<br>**1 Period is 50 minutes** | **Lectures per week** | **3** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |

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

**Books and References:**

| **Sr. No.** | **Title** | **Author/s** | **Publisher** | **Edition** | **Year** |
|---|---|---|---|---|---|
| 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 Practical

| B. Sc. (Information Technology) | | Semester – IV | |
|---|---|---|---|
| **Course Name: Software Engineering Practical** | | **Course Code: USIT4P4** | |
| **Periods per week**<br>**1 Period is 50 minutes** | **Lectures per week** | **3** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | | | |

| **List of Practical (To be executed using Star UML or any similar software)** | |
|---|---|
| **1.** | Study and implementation of class diagrams. |
| | |
| **2.** | Study and implementation of Use Case Diagrams. |
| | |
| **3.** | Study and implementation of Entity Relationship Diagrams. |
| | |
| **4.** | Study and implementation of Sequence Diagrams. |
| | |
| **5.** | Study and implementation of State Transition Diagrams. |
| | |
| **6.** | Study and implementation of Data Flow Diagrams. |
| | |
| **7.** | Study and implementation of Collaboration Diagrams. |
| | |
| **8.** | Study and implementation of Activity Diagrams. |
| | |
| **9.** | Study and implementation of Component Diagrams. |
| | |
| **10.** | Study and implementation of Deployment Diagrams. |
| | |

**Books and References:**

| Sr. No. | Title | Author/s | Publisher | Edition | Year |
|---|---|---|---|---|---|
| 1. | Object - Oriented Modeling and Design | Michael Blaha, James Rumbaugh | Pearson | | 2011 |
| 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 | Addison-Wesley | | 2005 |
| 4. | UML A Beginners Guide | Jason T. Roff | McGraw Hill Professional | | 2003 |

# Computer Graphics and Animation

| B. Sc. (Information Technology) | | Semester – IV | |
|---|---|---|---|
| **Course Name: Computer Graphics and Animation** | | **Course Code: USIT4P5** | |
| **Periods per week**<br>**1 Period is 50 minutes** | **Lectures per week** | **3** | |
| | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| | | | |

| **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.<br>i. Scaling about the origin followed by translation.<br>ii. Scaling with reference to an arbitrary point.<br>iii. Reflect about the line y = mx + c. |
| | |

| | |
|---|---|
| **8.** | **Solve the following:** |
| a. | Write a program to implement Cohen-Sutherland clipping. |
| b. | Write a program to implement Liang - Barsky Line Clipping Algorithm |
| | |
| **9.** | **Solve the following:** |
| a. | Write a program to fill a circle using Flood Fill Algorithm. |
| b. | Write a program to fill a circle using Boundary Fill Algorithm. |
| | |
| **10.** | **Solve the following:** |
| a. | Develop a simple text screen saver using graphics functions. |
| b. | Perform smiling face animation using graphic functions. |
| c. | Draw the moving car on the screen. |

**Books and References:**

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

**Evaluation Scheme:**

1. **Internal Evaluation (25 Marks).**
   i. **Test: 1 Class test of 20 marks. (Can be taken online)**

| Q | Attempt ***any four*** of the following: | 20 |
|---|---|---|
| a. | | |
| b. | | |
| c. | | |
| d. | | |
| e. | | |
| f. | | |

   ii. **5 marks: Active participation in the class, overall conduct, attendance.**

2. **External Examination: (75 marks)**

| | All questions are compulsory | |
|---|---|---|
| Q1 | (Based on Unit 1) Attempt ***any three*** of the following: | 15 |
| a. | | |
| b. | | |
| c. | | |
| d. | | |
| e. | | |
| f. | | |
| | | |
| Q2 | (Based on Unit 2) Attempt ***any three*** of the following: | 15 |
| Q3 | (Based on Unit 3) Attempt ***any three*** of the following: | 15 |
| Q4 | (Based on Unit 4) Attempt ***any three*** of the following: | 15 |
| Q5 | (Based on Unit 5) Attempt ***any three*** of the following: | 15 |

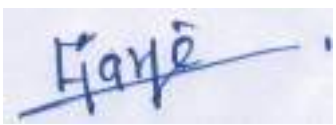
3. **Practical Exam: 50 marks**

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

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

**OR**

| | | |
|---|---|---|
| 1. | Practical Question | 40 |
| 2. | Journal | 5 |
| 3. | Viva Voce | 5 |



**Prof. Shivram S. Garje,**
**Dean**,
Faculty of Science and Technology