

**Hindi Vidya Prachar Samiti’s**

**Ramniranjan Jhunjhunwala College**

**of**

**Arts, Science & Commerce**

**(Autonomous College)**

**Affiliated to**

**UNIVERSITY OF MUMBAI**

**Syllabus for the S.Y.B.Sc.**

**Program: B.Sc. INFORMATION TECHNOLOGY Program Code: RJSUIT**

**DISTRIBUTION OF TOPICS AND CREDITS**

#### S.Y. B.Sc. INFORMATION TECHNOLOGY SEMESTER III

| **Course** | **Nomenclature** | **Credits** | **Topics** |
| --- | --- | --- | --- |
| RJSUIT301 | Object Oriented  Programming | 2 | 1. Object Oriented Principles  2. Introduction to Java  3. Exception Handling, Multithreading 4. File Handling  5. Packages |
| RJSUIT302 | Data Structures | 2 | 1. Introduction to DS  2. Arrays, LinkedList, Stacks, Queues, Trees, Graphs  3. Sorting, Searching and Hashing  techniques |
| RJSUIT303 | Computer Networks | 2 | 1. OSI and TCP/IP network model  2. Details of different layers  3. Wireless LAN |
| RJSUIT304 | Database Management Systems | 2 | 1. Introduction to databases and data models  2. Relational Database model, relational algebra and normalization  3. Constraints, views, Triggers  4. Transaction management and  Concurrency Control |
| RJSUIT305 | Applied Mathematics | 2 | 1. Matrices  2. Complex Numbers  3. Linear Differential Equations with Constant Coefficients  4. The Laplace Transform  5. Multiple Integrals  6. Beta and Gamma Functions  7. Error Function |
| RJSUITP301 | Object Oriented  Programming Practical | 2 |  |
| RJSUITP302 | Data Structures Practical | 2 |  |

| RJSUITP303 | Computer Networks  Practical | 2 |  |
| --- | --- | --- | --- |
| RJSUITP304 | Database Management Systems Practical | 2 |  |
| RJSUITP305 | Mobile Programming Practical | 2 |  |
| Total Credits |  | 20 |  |

#### S.Y. B.Sc. INFORMATION TECHNOLOGY SEMESTER IV

| **Course** | **Nomenclature** | **Credits** | **Topics** |
| --- | --- | --- | --- |
| RJSUIT401 | Advanced Java | 2 | 1. GUI Programming using JAVA 2. Database Connectivity using JDBC 3. Servlet, Cookies, Session object 4. Introduction to JSP  5. Networking with Java  6. Hibernate Programming |
| RJSUIT402 | Introduction to Embedded Systems | 2 | 1. Introduction to embedded system core  2. Types and components of  embedded systems  3. 8051 Microcontroller  4. Arduino uno |
| RJSUIT403 | Computer Oriented  Statistical Techniques | 2 | 1. Measures of central tendency and dispersion  2. Sampling, Decision and Estimation theory  3. Correlation theory |
| RJSUIT404 | Software Engineering | 2 | 1. Software Development Process Models, Agile S/W development  2. Requirements engineering processes 3. System models, Architectural  designs  4. Project management  5. Software testing |
| RJSUIT405 | Computer Graphics and Animation | 2 | 1. Introduction to Computer Graphics 2. Different drawing algorithms  3. Scaling, Translation, Reflection, Rotation and Shearing  4. Visible surface detection, viewing in 3D  5. Planes, Curves and surface.  6. Animation |

| RJSUITP401 | Advanced Java Practical | 2 |  |
| --- | --- | --- | --- |
| RJSUITP402 | Introduction to Embedded Systems Practical | 2 |  |
| RJSUITP403 | Computer Oriented  Statistical Techniques Practical | 2 |  |
| RJSUITP404 | Software Engineering  Practical | 2 |  |
| RJSUITP405 | Computer Graphics and Animation Practical | 2 |  |

| **B.Sc. (Information Technology)** | | Semester–III | |
| --- | --- | --- | --- |
| Object Oriented Programming | | **Course Code: RJSUIT301** | |
| **Lectures per week (1 Lecture is 50 minutes): Total Lectures:** | | **5**  **60** | |
| **Credits** | | **02** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
| **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lectures** |
| --- | --- | --- |
| **I** | **Object Oriented Methodology:**  Introduction, Advantages and Disadvantages of Procedure Oriented Languages, what is Object Oriented? What is Object Oriented Development? Object Oriented Themes, Benefits and Application of OOPS.  **Principles of OOPS:** OOPS Paradigm, Basic Concepts of OOPS: Objects, Classes, Data Abstraction and Data Encapsulation, Inheritance, Polymorphism, Dynamic Binding, Message Passing **Introduction to classes and Objects:** Class fundamentals, Declaring objects, Assigning object reference variables, Defining member functions, passing object as an argument, Returning object from functions, access specifier, static method, garbage collection. | **12** |
| **II** | **Introduction to Java:**  Java features, difference between Java and C++, Java Environment, Java Runtime Environment, Java development kit, Simple java program, Java program structure, Java tokens, Implementing A Java program, Java Virtual Machine, **Constants, Variables, Data types, operators and Expressions, Decision Making: branching and looping**  **ByteStreams:** InputStream, OutputStream, FileInputStream, FileOutputStream, CharacterStream: Reader, Writer, FileReader, File Writer, BufferedReader, BufferedWritter, PrintWriter | **12** |

| **III** | **Polymorphism:** Concept of function overloading, overloaded operators, Data Conversion between objects and basic types, constructors and destructors  **Inheritance:** Inheritance basics, defining derived classes, single inheritance, making private member inheritable, multilevel, multiple inheritance, hybrid, hierarchical inheritance, constructors in derived classes, limitations of using inheritance in java  **Virtual Functions:** Introduction and need, Pure Virtual Functions, **Abstract class** | **12** |
| --- | --- | --- |
| **IV** | **Interfaces:** Multiple inheritance in java, defining, extending and implementing interfaces. Accessing interface variables, Abstract methods and classes, final methods, variables and classes, this and super keywords.  **Multithreaded Programming:** creating threads, extending thread class, stopping and blocking a thread, lifecycle of thread, using thread method | **12** |
| **V** | **Managing Errors and Exceptions:** Types of errors, Exceptions, Syntax of Exception handling code, multiple catch statements, using finally statement, throwing our own exception, using exceptions for debugging.  **Packages:** Introduction, creating and accessing packages **Managing I/O files :** using the file class, creation of files, reading / writing characters, reading/ writing bytes, concatenating and buffering files, random access files | **12** |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| OOPs Practical | | **Course Code: RJSUIT3P1** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
| **Internal** | **--** | **--** |

| List of Practical: To be implemented using object-oriented language | |
| --- | --- |
| **1.** | **Classes and methods** |
| a. | Write a C++ / Java program to design an employee class for reading and displaying the employee information, the getInfo () and displayInfo() methods will be used respectively. |
| b. | (use C++ and Java programming) Write a C++ / Java program to design the class student containing getData() and displayData() as two of its methods which will be used for reading and displaying the student information respectively. Where getData() will be private method |
| c. | Write a Java Program to design the class Demo which will contain the following methods: readNo(), factorial() for calculating the factorial of a number, reverseNo() will reverse the given number, is Palindrome() will check the given number is palindrome, isArmstrong() which will calculate the given number is armStrong or not. Where readNo() will be a private method. |
| d. | Write a Java program to demonstrate function definition outside class and accessing class members in function definition. |
|  |  |
| **2.** | **Constructors and method overloading.** |
| a. | Write a C++ program to design a class “Complex” for adding the two complex numbers. Use a constructor. |
| b. | (Use C++) Write a Java Program to design a class Geometry containing the methods area () and volume() and also overload the area() function. |
| c. | Write a Java Program to demonstrate basic String handling |
| d. | Write a Java program to count the letters, spaces, numbers and other characters of an input string |
|  |  |
| **3.** | **Inheritance** |
| a. | Write a C++/ Java program to implement single level inheritance. |
| b. | Write a java program to implement method overriding |
| c. | Write a C++ / java program to implement multiple inheritance. |
| d. | Write a C++ program to implement hierarchical inheritance. |
|  |  |
| **4.** | **Use of operators:** |
| a. | Write a Java program to add two Binary Numbers. |
| b. | Write a Java program to convert decimal numbers to binary numbers and vice versa. |

|  |  |
| --- | --- |
| **5.** | **Array handling:** |
| a. | Write a Java program to sort a numeric array and a string array. |
| b. | Write a Java program to find the common elements between two arrays of |
| c. | integers. Write a java program to add two matrices and print the resultant matrix. |
| d. | Write a java program for multiplying two matrices and print the product for the |
|  | same. |
| **6.** | **Virtual functions and abstract classes** |
| a. | Write a C++ / Java program to implement the concept of method overriding. |
| b. | Write a java program to show the use of virtual function |
| c. | Write a java program to show the implementation of abstract class. |
|  |  |
| **7.** | **Multithreading** |
| a. | Write a java program to implement multithreading. |
| b. | Write a Java program to create multiple threads in Java. |
|  |  |
| **8.** | **Exception handling** |
| a. | Write a java program to show the implementation of exception handling |
| b. | Write a java program to show the implementation for exception handling for |
|  | strings |
| **9.** | **File handling** |
| a. | Write a java program to open a file and display the contents in the console window. |
| b. | Write a java program to copy the contents from one file to other file. |
| c. | Write a java program to read the student data from user and store it in the file. |
|  |  |
| **10.** | **Packages** |
| a. | Write a program to create a user defined package in Java. |
| b. | Write a Package MCA which has one class Student. Accept student details through |
| c. | Write a java program to copy the contents from one file to other file.parameterized constructor. Write display () method to display details. Create a |
| d. | Write a java program to read the student data from the user and store it in the file.main class which will use packages and calculate total marks and percentage. |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| Data Structures | | **Course Code: RJSUIT302** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect**  **ures** |
| --- | --- | --- |
| **I** | **Introduction:** Data and Information, Data Structure, Classification of Data Structures, Primitive Data Types, Abstract Data Types, Data structure vs. File Organization, Operations on Data Structure, Algorithm, Importance of Algorithm Analysis, Complexity of an Algorithm, Asymptotic Analysis and Notations, Big O Notation, Big Omega Notation, Big Theta Notation, Rate of Growth and Big O Notation.  **Array**: Introduction, One Dimensional Array, Memory Representation of One Dimensional Array, Traversing, Insertion, Deletion, Searching, Sorting, Merging of Arrays, Multidimensional Arrays, Memory Representation of Two Dimensional Arrays, General Multi- Dimensional Arrays, Sparse Arrays, 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**. Queue:** Introduction, Queue, Operations on the Queue, Memory Representation of Queue, Array representation of queue, Linked List Representation of Queue, Circular Queue, Some special kinds of queues, Deque, Priority Queue, Application of Priority Queue, Applications of Queues. | **12** |
| --- | --- | --- |
| **IV** | **Sorting and Searching Techniques**  Bubble, Selection, Insertion, Merge Sort. Searching: Sequential, Binary, Indexed Sequential Searches, Binary Search.  **Tree:** Tree, Binary Tree, Properties of Binary Tree, Memory Representation of Binary Tree, Operations Performed on Binary Tree, Reconstruction of Binary Tree from its Traversals, Huffman Algorithm, Binary Search Tree, Operations on Binary Search Tree, Heap, Memory Representation of Heap, Operation on Heap, Heap Sort.  **Advanced Tree Structures:** Red Black Tree, Operations Performed on Red Black Tree, AVL Tree, Operations performed on AVL Tree, 2-3 Tree, B-Tree. | **12** |
| **V** | **Hashing Techniques**  Hash function, Address calculation techniques, Common hashing functions Collision resolution, Linear probing, Quadratic, Double hashing, Bucket hashing, Deletion and rehashing  **Graph**: Introduction, Graph, Graph Terminology, Memory Representation of Graph, Adjacency Matrix Representation of Graph, Adjacency List or Linked Representation of Graph, Operations Performed on Graph, Graph Traversal, Applications of the Graph, Reachability, Shortest Path Problems, Spanning Trees. | **12** |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| DS Practical | | **Course Code: RJSUITP302** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| Practical  No | Details |
| --- | --- |
| **1.** | **Implement the following:** |
| a. | Write a program to store the elements in a 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 a 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 the 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. |
| 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. |
| **8.** | **Implement the following data structure techniques:** |
| a. | Write a program to insert the element into the maximum heap. |
| b. | Write a program to insert the element into the 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 the shortest path diagram. |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| Computer Networks | | **Course Code: RJSUIT303** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect**  **ures** |
| --- | --- | --- |
| **I** | **Introduction:** Data communications, networks, network types, Internet history, standards and administration.  **Network Models:** Protocol layering, TCP/IP protocol suite, The OSI model. **Introduction to Physical layer:** Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance. **Digital and Analog transmission:** Digital-to-digital conversion, analog-to digital conversion, transmission modes, digital-to-analog conversion, analog-to analog conversion. | **12** |
| **II** | **Bandwidth Utilization: Multiplexing and Spectrum Spreading:** Multiplexing, Spread Spectrum  **Transmission media:** Guided Media**,** Unguided Media  **Switching:** Introduction, circuit switched networks, packet switching, and structure of a switch.  **Introduction to the Data Link Layer:** Link layer addressing, Data Link Layer Design Issues, Error detection and correction, block coding, cyclic codes, checksum, forward error correction, error correcting codes, error detecting codes. | **12** |
| **III** | **Data Link Control:** DLC services, data link layer protocols, HDLC, Point-to point protocol.  **Media Access Control**: Random access, controlled access, channelization, Wired LANs – Ethernet Protocol, standard ethernet, fast ethernet **Wireless LANs:** Introduction, IEEE 802.11 project, Bluetooth, Cellular | **12** |

|  | telephony, Satellite networks.  **Connecting devices** |  |
| --- | --- | --- |
| **IV** | **Introduction to the Network Layer**: Network layer services, network layer performance, IPv4 addressing, forwarding of IP packets, Internet Protocol, ICMPv4, Mobile IP  **Unicast Routing:** Introduction, routing algorithms, unicast routing protocols. **Next generation IP:** IPv6 addressing, IPv6 protocol, transition from IPv4 to IPv6. | **12** |
| **V** | **Introduction to the Transport Layer**: Introduction, Transport layer protocols, Sliding Window Protocol, Go-Back-n protocol, Selective repeat protocol, Bidirectional protocols), Transport layer services, User datagram protocol, Transmission control protocol, **Standard Client0Server Protocols:** World wide-web and HTTP, FTP, Electronic mail, Domain name system. | **12** |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| CN Practical | | **Course Code: RJSUITP303** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| **Practical No** | **Details** |
| --- | --- |
| **1.** | **IPv4 Addressing and Subnetting**  a) Given an IP address and network mask, determine other information about the IP address such as:  ● Network address  ● Network broadcast address  ● Total number of host bits  ● Number of hosts  b) Given an IP address and network mask, determine other information about the IP address such as:  ● The subnet address of this subnet  ● The broadcast address of this subnet  ● The range of host addresses for this subnet  ● The maximum number of subnets for this subnet mask  ● The number of hosts for each subnet  ● The number of subnet bits  ● The number of this subnet |
| **2.** | Use of ping and tracert / traceroute, ipconfig / ifconfig, route and arp utilities. |
| **3.** | Configure IP static routing. |
| **4.** | Configure IP routing using RIP. |
| **5.** | Configuring Simple OSPF. |
| **6.** | Configuring DHCP server and client. |
| **7.** | Create a virtual PC based network using virtualization software and virtual NIC. |
| **8.** | Configuring DNS Server and client. |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| Database Management Systems | | **Course Code: RJSUIT304** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect ures** |
| --- | --- | --- |
| **I** | **Introduction to Databases**  What is database system, purpose of database system, view of data, relational databases, database architecture, Database administrator, Role of DBA **Data Models:**  The importance of data models, Basic building blocks, Business rules, The evolution of data models, Degrees of data abstraction.  **ER Diagram and Database Schema:**  Database design and E R Model: overview, ER Model: Entity set, Relationship set, Attributes, Types of attributes, Degree of relationship, Mapping Constraints, relationship participation, E R Diagrams, E R D Issues, weak entity sets, Codd’s rules, Reduction of ER Schema to Tables | **12** |
| **II** | **Relational database model:**  Logical view of data, keys, integrity rules, Relational Database design: features of good relational database design, atomic domain and Normalization (1NF, 2NF, 3NF, BCNF).  **Relational Algebra and Calculus Relational algebra:** Introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison. **Calculus**: Tuple relational calculus, Domain relational Calculus, calculus vs algebra, computational capabilities | **12** |

| **III** | **Constraints, Views and SQL:** Constraints, types of constraints, Integrity constraints, Views: Introduction to views, data independence, security, updates on views, comparison between tables and views SQL: data definition, aggregate function, Null Values, nested sub queries, Joined relations. Triggers. | **12** |
| --- | --- | --- |
| **IV** | **Transaction management and Concurrency Control**  Control Transaction management: ACID properties, serializability and concurrency control, Lock based concurrency control (2PL, Deadlocks), Time stamping methods, optimistic methods, database recovery management. | **12** |
| **V** | **PL-SQL:** Beginning with PL / SQL, Identifiers and Keywords, Operators, Expressions, Sequences, Control Structures, Cursors and Transaction, Collections and composite data types, Procedures and Functions, Exceptions Handling, Packages, With Clause and Hierarchical Retrieval, Triggers | **12** |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| DBMS Practical | | **Course Code: RJSUITP304** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| **Practical No** | **Details** |
| --- | --- |
| **1.** | **Creating and Managing Tables** |
| a. | Creating and Managing Tables |
| b. | Including Constraints |
| **2.** | **Manipulating Data** |
| a. | Using INSERT statement |
| b. | Using DELETE statement |
| c. | Using UPDATE statement |
| **3.** | **SQL Statements – 1** |
| a. | Writing Basic SQL SELECT Statements |
| b. | Restricting and Sorting Data |
| c. | Single-Row Functions |
| **4.** | **SQL Statements – 2** |
| a. | Displaying Data from Multiple Tables |
| b. | Aggregating Data Using Group Functions |
| c. | Subqueries |
| **5.** | **Creating and managing other database objects** |
| a. | Creating Views |
| b. | Other Database Objects |
| c. | Controlling User Access |

| **6.** | **Using SET operators, Date/Time Functions, GROUP BY clause (advanced features) and advanced subqueries** |
| --- | --- |
| a. | Using SET Operators |
| b. | Datetime Functions |
| c. | Enhancements to the GROUP BY Clause |
| d. | Advanced Subqueries |
| **7.** | **PL/SQL Basics** |
| a. | Declaring Variables |
| b. | Writing Executable Statements |
| c. | Interacting with the Oracle Server |
| d. | Writing Control Structures |
| **8.** | **Composite data types, cursors and exceptions.** |
| a. | Working with Composite Data Types |
| b. | Writing Explicit Cursors |
| c. | Handling Exceptions |
| **9.** | **PL/SQL - Procedures and Functions** |
| a. | Creating Procedures |
| b. | Creating Function |
| **10.** | **PL/SQL - Creating Database Triggers** |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| Applied Mathematics | | **Course Code: RJSUIT305** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **02** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect**  **ures** |
| --- | --- | --- |
| **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.  **Introduction to 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 | **12** |
| **II** | **Complex Numbers:** 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)  **Introduction to Application of Derivatives**  **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.  **Linear Differential Equations with Constant Coefficients:** Introduction, The Differential Operator, Linear Differential Equation f(D) y = 0, Different cases | **12** |

|  | 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** | **Introduction to Partial derivatives**  **Multiple Integrals:** Double Integral, Change of the order of the integration, Double integral in polar co-ordinates, Triple integrals.  **Applications of integration:** Areas, Volumes of solids. | **12** |
| **V** | **Beta and Gamma Functions –** Definitions**,** Properties and Problems. Duplication formula.  **Differentiation Under the Integral Sign:** Introduction, DUIS with one parameter, Leibniz theorem for DUIS  **Error Functions:** Definition, Complimentary error function definition, Properties and Problems | **12** |

| **B.Sc. (Information Technology)** | | **Semester–III** | |
| --- | --- | --- | --- |
| Mobile Programming Practical | | **Course Code: RJSUITP305** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| **Practical No** | **Details** |
| --- | --- |
| **1** | **Introduction to App Inventor and App Inventor IDE**  **Introduction to User Interface components**  (i) Design an app to demonstrate use of button, label and notifier component (ii) Design an app to demonstrate the use of math and logic block (iii) Design a sequence calculator app using of slider component |
| **2** | **Introduction to Layout component**  (i) Design an app for the Kilometer Converter  (ii) Design an app to calculate Test Average  (iii) Design an app with layout components and Color Blocks |
| **3** | **Introduction to Sensor component**  (i) Design an app that demonstrates the use of TexttoSpeech component (ii) Design an app that demonstrates the use of AccelerometerSensor component (iii) Design an app that demonstrates the use of SpeechRecognizer component |
| **4** | **Introduction to Media component**  (i) Design an app that uses the Camcorder component to capture a video and play it back using the VideoPlayer component  (ii) Design an app using camera component to capture image |
| **5** | **Introduction to Drawing and Animation component**  (i) Design an app that demonstrates the use of canvas component (ii) Design an app that demonstrates the use of a ball and an ImageSprite to simulate a golf game |

| **6** | **Introduction to Map component and social component**  (i) Design an app that demonstrates the use ContactPicker and  PhoneNumberPicker component  (ii) Design an app that demonstrates the use of Map component |
| --- | --- |
| **7** | **Introduction to storage component**  (i) Design an app uses the PhoneNumberPicker and a TinyDB to store a contact in a list and make a phone call to a contact from that list. |
| **8** | (i) Design a Game Swat a Mosquito  (ii) Design an app that can be used as a replacement for a physical dice (iii) Design a Scratch card with App Inventor |

**Commerce S.Y. B.Sc. INFORMATION TECHNOLOGY Syllabus Semester III & IV**

| **B.Sc. (Information Technology)** | | Semester–IV | |
| --- | --- | --- | --- |
| Advanced Java Programming | | **Course Code: RJSUIT401** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect**  **ures** |
| --- | --- | --- |
| **I** | **Introduction to GUI:**  **Applet:** Lifecycle of java applet, creating an applet, applet tag attributes, passing parameters to applet  **AWT:** Window Fundamentals, basic UI components, Layouts  **Event Handling:** event driven programs, delegating the event, java event type, event classes | **12** |
| **II** | **Understanding Java EE:** What is an Enterprise Application? What is java enterprise edition? Java EE Technologies, Java EE evolution, Glassfish server **Java EE Architecture, Server and Containers:** Types of System Architecture, Java EE Server, Java EE Containers.  **Introduction to Java Servlets:** The Need for Dynamic Content, Java Servlet Technology, Why Servlets? What can Servlets do?  **Servlet API and Lifecycle:** Java Servlet API, The Servlet Skeleton, The Servlet Life Cycle, A Simple Welcome Servlet  **Working With Servlets:** Getting Started, Using Annotations Instead of Deployment Descriptor. | **12** |

| **III** | **Working with Databases:** What Is JDBC? JDBC Architecture, Accessing Database, The Servlet GUI and Database Example.  **COOKIES:** Kinds of Cookies, Where Cookies Are Used? Creating Cookies Using Servlet, Dynamically Changing the Colors of a Page  **SESSION:** What Are Sessions? Lifecycle Of Http Session, Session Tracking with Servlet API, A Servlet Session Example  **Request Dispatcher:** Request dispatcher Interface, Methods of Request Dispatcher, Request Dispatcher Application. | **12** |
| --- | --- | --- |
| **IV** | **Introduction To Java Server Pages:** Why use Java Server Pages? Disadvantages Of JSP, JSP v\s Servlets, Life Cycle of a JSP Page, how does a JSP function? How does JSP execute? About Java Server Pages. **Getting Started with Java Server Pages:** Comments, JSP Document, JSP Elements, JSP GUI Example. **Action Elements:** Including other Files, Forwarding JSP Page to Another Page, Passing Parameters for other Actions, Loading a Javabean.  **Implicit Objects, Scope and El Expressions:** Implicit Objects, Character Quoting Conventions, Unified Expression Language [Unified El], Expression Language. **Java Server Pages Standard Tag Libraries:** What is wrong in using JSP Scriptlet Tags? How JSTL Fixes JSP Scriptlet's Shortcomings? Disadvantages Of JSTL, Tag Libraries. | **12** |
| **V** | **Networking in Java:** socket and networking, socket overview, proxy server, internet addressing, DNS, using UDP connection, using TCP connection, working with URL  **Introduction to Hibernate:** What is Hibernate? Why Hibernate? Hibernate, Database and The Application, Components of Hibernate, Architecture of Hibernate, How Hibernate Works?  **Writing Hibernate Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database and Tables in Mysql, creating a Web Application, Adding The Required Library Files, Creating a Javabean Class, Creating Hibernate Configuration File, Adding a Mapping Class, Creating JSPS, Running The Hibernate Application. |  |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Advanced Java Practical | | **Course Code: RJSUITP401** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| **Practical**  **No** | **Details** |
| --- | --- |
| **1.** | **Applet Programming** |
| a. | Write a Java program to demonstrate lifecycle of an applet |
| **2.** | **GUI Programming** |
| a. | Design an AWT program to print the factorial for an input value. |
| b. | Design an AWT program to perform various string operations like reverse string, string concatenation etc. |
| **3.** | **Layouts and Event Handling** |
| a. | Design an AWT application to demonstrate different layouts |
| b. | Write a java program to implement –  Text Events  Mouse Events  Windows Event |
| **4.** | **Introduction to Servlet Programming** |
| a. | Create a simple calculator application using servlet. |
| b. | Create a servlet for a login page. If the username and password are correct then it says message “Hello <username>” else a message “login failed” |
| **5.** | **Servlet Programming with Cookies and Session** |
| a. | Using Request Dispatcher Interface create a Servlet which will validate the password entered by the user, if the user has entered "Servlet" as password, then he will be forwarded to Welcome Servlet else the user will stay on the index.html page and an error message will be displayed. |

| b. | Create a servlet that uses Cookies to store the number of times a user has visited servlet. |
| --- | --- |
| c. | Create a servlet demonstrating the use of session creation and destruction. Also check whether the user has visited this page first time or has visited earlier also using sessions. |
| **6.** | **Java Database Connectivity** |
| a. | Write a servlet program to demonstrate simple database connectivity |
| b. | Develop a Simple Servlet Question Answer Application using Database. |
| **7.** | **JSP** |
| a. | Develop a simple JSP application to display values obtained from the use of intrinsic objects of various types. |
| b. | Develop a simple JSP application to pass values from one page to another with validations. (Name-txt, age-txt, hobbies-checkbox, email-txt, gender-radio button). |
| **8.** | **JSP with Database Connectivity** |
| a. | Write a java program to generate a JSP application to demonstrate database connectivity. |
| b. | Generate a Guestbook application using taglib |
| **9.** | **Networking in Java** |
| a. | Create a socket program to implement TCP protocol |
| b. | Write a java program to implement UDP protocol |
| **10.** | **Implement the following Hibernate applications.** |
| a. | Develop an application to demonstrate Hibernate One- To -One Mapping Using Annotation. |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Introduction to Embedded Systems | | **Course Code: RJSUIT402** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect**  **ures** |
| --- | --- | --- |
| **I** | **Introduction:** Embedded Systems and general-purpose computer systems, history, classifications, applications and purpose of embedded systems **Core of embedded systems:** microprocessors and microcontrollers, RISC and CISC controllers, Big endian and Little-endian processors, Application specific ICs, Programmable logic devices, COTS, sensors and actuators, communication interface, embedded firmware, other system components.  **Characteristics and quality attributes of embedded systems:** Characteristics, operational and non-operational quality attributes. | **12** |
| **II** | **Embedded Systems – Application and Domain Specific:** Application specific – washing machine, domain specific - automotive.  **Peripherals:** Control and Status Registers, Device Driver, Timer Driver - Watchdog Timers.  **Programming embedded systems:** structure of embedded program, infinite loop, compiling, linking and debugging. | **12** |
| **III** | **The 8051 Microcontrollers:** Microcontrollers and Embedded processors, Overview of 8051 family.8051 Microcontroller hardware, Input/output pins, Ports, and Circuits, External Memory.  **Designing Embedded System with 8051 Microcontroller:** Factors to be considered in selecting a controller, why 8051 Microcontroller, Designing with 8051.  **8051 Programming in C:** | **12** |

|  | Data Types and time delay in 8051 C, I/O Programming, Logic operations, Data conversion Programs. |  |
| --- | --- | --- |
| **IV** | **Introduction to Arduino**: Arduinouno, Arduino Mega and Arduino Nano, Arduino and display devices, Arduino and digital input output devices (push button), Arduino and motors  **Arduino Programming:** Arduino IDE, Installing and launching the IDE, program structure, Data types, variables, constants, control structure and loops, Functions  **Arduino and wireless communication:** RF modem, global system for mobile modem | **12** |
| **V** | **Real Time Operating System (RTOS):** Operating system basics, types of operating systems, Real-Time Characteristics, Selection Process of an RTOS. **Design and Development:** Embedded system development Environment – IDE, types of files generated on cross compilation, disassembler/ de-compiler, simulator, emulator and debugging, embedded product development life-cycle, trends in embedded industry | **12** |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Introduction to Embedded Systems Practical | | **Course Code: RJSUITP402** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| **Practical**  **No** | | **Details** |
| --- | --- | --- |
| **1.** |  | Demonstrate the use of reprogrammable computer to perform Programming, Execution and debugging of embedded program using -  a. 8051 microcontrollers  b. Arduinouno |
| **2.** | a. | To demonstrate use of general-purpose port i.e. Input/ output port of two 8051 controllers for data transfer between them. |
|  | b. | Demonstrate use of arduino ports using push button and LED |
| **3.** | a. | Port I / O: Use one of the four ports of 8051 for O/P interfaced to eight LED’s. Simulate binary counter (8 bit) on LED’s |
|  | b. | To interface 8 LEDs at Input-output ports of 8051 microcontroller and create different patterns. |
| **4.** | a. | Serial I / O: Configure 8051 serial port for asynchronous serial communication with serial port of PC exchange text messages to PC and display on PC screen. Signify end of message by carriage return. |
|  | b. | To demonstrate interfacing of seven-segment LED display with 8051 microcontroller and generate counting from 0 to 99 with fixed time delay. |
| **5.** | a. | Interface 8051 with D/A converter and generate square wave of given frequency on oscilloscope. |
|  | b. | Interface 8051 with D/A converter and generate triangular wave of given frequency on oscilloscope. |
|  | c. | Using D/A converter generate sine wave on oscilloscope with the help of lookup table stored in data area of 8051. |
| **6.** |  | Interface stepper motor with 8051 and write a program to move the motor |

|  |  | through a given angle in clock wise or counter clock wise direction. |
| --- | --- | --- |
| **7.** |  | Generate traffic signal using 8051 microcontrollers |
| 8. |  | Implement temperature controller using 8051 |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Computer Oriented Statistical Techniques | | **Course Code: RJSUIT403** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect**  **ures** |
| --- | --- | --- |
| **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.  **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.  **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.  **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.  **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 Di 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.  **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.  **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.  **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 Contingency, Correlation of Attributes, Additive Property of chi- square. | **12** |
| --- | --- | --- |
| **V** | **Curve Fitting and the Method of Least Squares:** Relationship Between Variables, Curve Fitting, Equations of Approximating Curves, Freehand Method of Curve Fitting, The Straight Line, The Method of Least Squares, The Least Squares Line, Nonlinear Relationships, The Least-Squares Parabola, Regression, Applications to Time Series, Problems Involving More Than Two Variables.  **Correlation Theory:** Correlation and Regression, Linear Correlation, Measures of Correlation, The Least-Squares Regression Lines, Standard Error of Estimate, Explained and Unexplained Variation, Coefficient of correlation, Remarks concerning the Correlation coefficient, Product-Moment Formula for the Linear Correlation Coefficient, Short Computational Formulas, Regression Lines and the Linear Correlation Coefficient, Correlation of Time  Series, Correlation of Attributes, Sampling Theory of Correlation, Sampling Theory of Regression. | **12** |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Computer Oriented Statistical Techniques Practical | | **Course Code: RJSUITP403** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| **Practical No** | **Details** |
| --- | --- |
| **1.** | Using R execute the basic commands, array, list and frames. |
| **2.** | Create a Matrix using R and Perform the operations addition, inverse, transpose and multiplication operations. |
| **3.** | Using R Execute the statistical functions: mean, median, mode, quartiles, range, inter quartile range histogram |
| **4.** | Using R import the data from Excel / .CSV file and perform the above functions. |
| **5.** | Using R import the data from Excel / .CSV file and calculate the standard deviation, variance, co-variance. |
| **6.** | Using R import the data from Excel / .CSV file and draw the skewness. |
| **7.** | Import the data from Excel / .CSV and perform the hypothetical testing. |
| **8.** | Import the data from Excel / .CSV and perform the Chi-squared Test. |
| **9.** | Perform the Linear Regression using R. |
| **10.** | Compute the Least squares means using R |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Software Engineering | | **Course Code: RJSUIT404** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect**  **ures** |
| --- | --- | --- |
| **I** | **Introduction:** What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc. **Software Development Process Models.**  ● Waterfall Model.  ● Prototyping.  ● Iterative Development.  ● Rational Unified Process.  **Agile software development**: Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.  **Requirements Engineering Processes:** Feasibility study, Requirement elicitation and analysis, Requirements Validations, Requirements Management. | **12** |
| **II** | **System Models:** Models and its types, Context Models, Behavioural Models, Data Models, Object Models, Structured Methods.  **Architectural Design:** Architectural Design Decisions, System Organisation, Modular Decomposition Styles, Control Styles, Reference Architectures. **User Interface Design:** Need of UI design, Design issues, The UI design Process, User analysis, User Interface Prototyping, Interface Evaluation. | **12** |

| **III** | **Project Management**  Software Project Management, Management activities, Project Planning, Project Scheduling, Risk Management.  **Quality Management:** Process and Product Quality, Quality assurance and Standards, Quality Planning, Quality Control, Software Measurement and Metrics.  **Process Improvement:** Process and product quality, Process Classification, Process Measurement, Process Analysis and Modeling, Process Change, The CMMI Process Improvement Framework. | **12** |
| --- | --- | --- |
| **IV** | **Verification and Validation:** Planning Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods. **Software Testing:** System Testing, Component Testing, Test Case Design, Test Automation.  **Software Cost Estimation:** Software Productivity, Estimation Techniques, Algorithmic Cost Modelling, Project Duration and Staffing | **12** |
| **V** | **Service Oriented Software Engineering:** Services as reusable components, Service Engineering, Software Development with Services.  **Software reuse:** The reuse landscape, Application frameworks, Software product lines, COTS product reuse.  **Distributed software engineering**: Distributed systems issues, Client–server computing, Architectural patterns for distributed systems, Software as a service | **12** |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Software Engineering Practical | | **Course Code: RJSUITP404** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| **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. |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Computer Graphics and Animation | | **Course Code: RJSUIT405** | |
| **Lectures per week (1 Lecture is 50 minutes)**  **Total Lectures** | | **5**  **60** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Theory Examination** | **2** | **60** |
|  | **Internal** | **--** | **40** |

| **Unit** | **Details** | **Lect**  **ures** |
| --- | --- | --- |
| **I** | **Introduction to Computer Graphics:**  Overview of Computer Graphics, Computer Graphics Application and Software, Description of some graphics devices, Input Devices for Operator Interaction, Active and Passive Graphics Devices, Display Technologies, Video Display Devices, Cathode Ray Tube Basics, Raster Refresh (Raster  Scan) Displays, Random-Scan Display, Color CRT Monitors, Graphics Monitors and Workstations.  **Scan conversion:**  Digital Differential Analyzer (DDA) algorithm, Bresenham’s Line drawing algorithm. Bresenham’s method of Circle drawing, Midpoint Circle Algorithm, Midpoint Ellipse Algorithm, Boundary-Fill algorithm, Flood-Fill algorithm | **12** |
| **II** | **Two-Dimensional Transformations:**  Basic Transformations: Translation, Rotation, Scaling. Transformations and Matrices, Transformation Conventions, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Composite Transformation: Translations, Rotation, Reflection, Scaling, Shearing , Transformation of Points, Transformation of The Unit Square, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations, Clipping Operations: Line Clipping algorithms–Cohen-Sutherland and Liang- |  |

|  | Barsky, Polygon Clipping Algorithms- Sutherland Hodgeman and Weiler Atherton, Curve Clipping, Text Clipping.  **Three-Dimensional Transformations:**  Three-Dimensional Translation, Three-Dimensional Scaling, Three Dimensional Shearing, Three-Dimensional Rotations, Three-Dimensional Reflection, Matrix Representation of 3D Transformations, Composite 3D Transformations. |  |
| --- | --- | --- |
| **III** | **Viewing in 3D**  Viewing Pipeline, Viewing Coordinates, Examples of 3D Viewing, Projections: Parallel Projections, Perspective Projections, Orthographic Projections, Oblique Projections, View Volumes and General Projection Transformations, Viewport Clipping, Clipping in Homogeneous Coordinates.  **Light:** Basic Light Sources, Basic Illumination Models: Ambient Light, Diffuse Reflection, Intensity Attenuation, Transparency, Shadows. **Color Models and Color Applications:** Chromaticity Diagram, Color Models. | **12** |
| **IV** | **Visible-Surface Determination:**  Classification of Visible-Surface Detection Algorithms, Back-Face Detection, Depth-Buffer Method, A-Buffer Method, Scan-Line Method, Painter’s algorithms, BSP tree Method, Curved Surfaces, Wireframe Methods **Plane Curves and Surfaces:**  Curve Representation, Spline Representation, Cubic Splines, Bezier Curves, B-spline Curves, B-spline Curve Fit, B-spline Curve Subdivision, Parametric Cubic Curves, and Quadric Surfaces. Bezier Surfaces | **12** |
| **V** | **Computer Animation:**  Principles of Animation, General Computer Animation Functions, Key frame Systems, Morphing, Character Animation, Physics-Based Animation, Procedural Techniques.  **Image Manipulation and Storage:**  What is an Image? Digital image file formats, Image compression standard– JPEG, Image Processing-Digital image enhancement, contrast stretching, Histogram Equalization, smoothing and median Filtering. | **12** |

| **B.Sc. (Information Technology)** | | **Semester–IV** | |
| --- | --- | --- | --- |
| Computer Graphics and Animation Practical | | **Course Code: RJSUITP405** | |
| **Lectures per week (1 Lecture is 50 minutes)** | | **3** | |
| **Credits** | | **2** | |
|  | | **Hours** | **Marks** |
| **Evaluation System** | **Practical Examination** | **2½** | **50** |
|  | **External** | **2--** | **--** |

| **Practical No** | **Details** |
| --- | --- |
| **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 regions, 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 ellipse drawing algorithm. |
| **6.** | **Solve the following:** |
| a. | Write a program to implement 2D scaling. |
| b. | Write a program to perform 2D translation |

| **7.** | **Solve the following:** |
| --- | --- |
| a. | Perform 2D Rotation on a given object. |
| b. | Program to create a house like figure and perform the following operations. i. Scaling about the origin followed by translation.  ii. Scaling with reference to an arbitrary point.  iii. Reflect about the line y = mx + c. |
| **8.** | **Solve the following:** |
| a. | Write a program to implement Cohen-Sutherland clipping. |
| b. | Write a program to implement Liang - Barsky Line Clipping Algorithm |
| **9.** | **Solve the following:** |
| a. | Write a program to fill a circle using Flood Fill Algorithm. |
| b. | Write a program to fill a circle using Boundary Fill Algorithm. |
| **10.** | **Solve the following:** |
| a. | Develop a simple text screen saver using graphics functions. |
| b. | Perform smiling face animation using graphic functions. |

**Scheme of Examinations**

1. Two Internals of 20 marks each. One internal of MCQ format, 20 questions, 20 marks of Duration 30 min. Second Internal based on continuous evaluation on (Assignment/Quiz/ Presentation)

2. One External (Semester End Examination) of 60 marks. Duration 2 hours. 3. Practical Examination for each subject at the end of Semester. Total five practical components, each subject 50 marks with separate passing out of 50.

4. Minimum marks for passing the Theory and Practical Exam is 40 %.

5. Students must appear for at least one of the two Internal Tests to be eligible for the Semester End Examination.

6. For any KT examinations, there shall be ODD-ODD/EVEN-EVEN pattern followed. 7. A candidate will be allowed to appear for the practical examinations if he/she submits a certified journal of S.Y. B.Sc. Information Technology or a certificate from the Coordinator / Head of the Institute to the effect that the candidate has completed the practical course of S.Y. B.Sc. Information Technology as per the minimum requirements.

8. In case of loss of a journal, a candidate must produce a certificate from the Head of the department /Institute that the practical for the academic year were completed by the student. However, such a candidate will be allowed to appear for the practical examination, but the marks allotted for the journal will not be granted.

9. Decision of the coordinator, in consultation with the principal shall remain final and abiding to all.

**Evaluation and Assessment**

**Evaluation (Theory): Total marks per course - 100.**

**CIA- 40 marks**

CIA 1: Written test -20 marks

CIA 2: Written Test / Assignment / Quiz / Mini project -20 marks

**Semester End Examination – 60 marks**

Question paper covering all units

| **Question** | **Knowledge/Understanding** | **Application/Analysis** | **Total Marks**  **Per Unit** |
| --- | --- | --- | --- |
| **Unit 1** | **06** | **06** | **12** |
| **Unit 2** | **06** | **06** | **12** |
| **Unit 3** | **06** | **06** | **12** |
| **Unit 4** | **06** | **06** | **12** |
| **Unit 5** | **06** | **06** | **12** |
| **Total – Per**  **Objective** | **30** | **30** | **60** |
| **% Weightage** | **50** | **50** | **100%** |

**Evaluation of Practical: Total marks per course – 50**

• Continuous evaluation based on regular hands-on work submitted by students 30%

• Course End Practical Evaluation based on understanding of problem statement and implementation skill of students.

**Mapping of the courses to Employability / Entrepreneurship / Skill development SEMESTER III**

| **Course Code** | **Course Name** | **Unit No. And topics focusing on Employability / Entrepreneurship / Skill development** |
| --- | --- | --- |
| RJSUIT301 | Object Oriented Programming | **Core Course/Skill Enhancement:**  **Unit-I:** Principles of OOPS: OOPS, Introduction to classes and Objects.  **Unit-II:** Introduction to JAVA Basic Concepts **Unit -III:** Polymorphism, Inheritance  **Unit - IV:** Interfaces, Multi-threaded programming. **Unit- V:** Exception handling, Packages, Managing I/O files. |
|  | RJSUIT302 Data Structures | **Skill Enhancement**  **Unit I:** Introduction, Array  **Unit II:** Linked List  **Unit III:** Stack, Queue  **Unit IV:** Sorting and Searching Techniques, Tree, Advanced Tree Structures  **Unit V:** Hashing Techniques, Graph |
| RJSUIT303 | Computer  Networks | **Skill Enhancement, Employability**  **Unit I:** Introduction, Network Models, Introduction to Physical layer, Digital and Analog transmission **Unit II:** Bandwidth Utilization: Multiplexing and Spectrum Spreading, Transmission media, Switching, Introduction to the Data Link Layer  **Unit III:** Data Link Control, Media Access Control, Wireless LANs, Connecting devices  **Unit IV:** Introduction to the Network Layer, Unicast Routing, Next generation IP  **Unit V:** Introduction to the Transport Layer, Standard Client Server Protocols |
| RJSUIT304 | Database  Management  Systems | **Skill Enhancement, Employability**  **Unit I:** Introduction to Databases, Data Models, ER Diagram and Database Schema.  **Unit II:** Relational database model, Relational Algebra and Calculus Relational algebra, Calculus  **Unit III:** Constraints, Views and SQL  **Unit IV:** Transaction management and Concurrency **Unit V:** PL-SQL |
| RJSUIT305 | Applied  Mathematics | **Skill Enhancement**  **Unit I:** Matrices, Complex Numbers  **Unit II:** Equation of the first order and of the first degree, Linear Differential Equations with Constant Coefficients  **Unit III:** The Laplace Transform, Inverse Laplace |

|  |  | Transform  **Unit IV:** Multiple Integrals, Applications of integration **Unit V:** Beta and Gamma Functions, Differentiation Under the Integral Sign, Error Functions |
| --- | --- | --- |
| RJSUITP301 | Object Oriented Programming  Practical | **Skill Enhancement**  Object Oriented Programming with JAVA Practical |
| RJSUITP302 | Data Structures Practical | **Skill Enhancement** |
| RJSUITP303 | Computer Networks Practical | **Skill Enhancement** |
| RJSUITP304 | Database  Management  Systems Practical | **Skill Enhancement, Employability** |
| RJSUITP305 | Mobile  Programming  Practical | **Skill Enhancement**  Android development using App Inventor and App Inventor IDE |

**Mapping of the courses to Employability / Entrepreneurship / Skill development SEMESTER IV**

| **Course Code** | **Course Name** | **Topics** |
| --- | --- | --- |
| RJSUIT401 | Advanced Java | **Skill Enhancement, Employability**  **Unit -I:** Introduction- Applet, AWT, Event Handling  **Unit - II:** Java EE Architecture, Server and Containers, Working with Servlet  **Unit - III:** Working with databases (JDBC) **Unit - IV:** Introduction to Java Server Pages **Unit - V:** Networking in Java, Introduction to Hibernate |
| RJSUIT402 | Introduction to  Embedded Systems | **Skill Enhancement, Employability**  **Unit I:** Introduction, Core of embedded systems, **Unit II:** Embedded Systems – Application and Domain Specific, Embedded Hardware, Peripherals **Unit III:** The 8051 Microcontrollers, 8051 Programming in C  **Unit IV:** Arduino Programming  **Unit V:** Real Time Operating System (RTOS), Design and Development |
| RJSUIT403 | Computer Oriented Statistical  Techniques | **Skill Enhancement**  **Unit I:** The Mean, Median, Mode, and Other Measures of Central Tendency, The Standard Deviation and Other Measures of Dispersion, Introduction to R  **Unit II:** Moments, Skewness, and Kurtosis, Elementary Probability Theory, Elementary Sampling Theory  **Unit III:** Statistical Estimation Theory, Statistical Decision Theory, Statistics in R  **Unit IV:** Small Sampling Theory, The Chi-Square Test  **Unit V:** Curve Fitting and the Method of Least Squares, Correlation Theory |
| RJSUIT404 | Software Engineering | **Skill Enhancement**  **Unit I:** Introduction, Software Requirements, Software Processes, Software Development Process Models, Agile software development  **Unit II:** Socio-technical system, Critical system, Requirements Engineering Processes, System Models  **Unit III:** Architectural Design, User Interface Design, Project Management, Quality Management |

|  |  | **Unit IV:** Verification and Validation, Software Testing, Software Measurement, Software Cost Estimation  **Unit V:** Process Improvement, Service Oriented Software Engineering, Software reuse, Distributed software engineering |
| --- | --- | --- |
| RJSUIT405 | Computer Graphics and Animation | **Skill Enhancement**  **Unit I:** Scan conversion  **Unit II:** Two-Dimensional Transformations, Three Dimensional Transformations  **Unit III:** Viewing in 3D, Light, Color Models and Color Applications  **Unit IV:** Visible-Surface Determination, Plane Curves and Surfaces  **Unit V:** Computer Animation, Image Manipulation and Storage |
| RJSUITP401 | Advanced Java  Practical | **Skill Enhancement**  Practical on all concepts in Advanced Java. |
| RJSUITP402 | Introduction to  Embedded Systems Practical | **Skill Enhancement, Employability**  Arduino Programming |
| RJSUITP403 | Computer Oriented Statistical Techniques Practical | **Skill Enhancement** |
| RJSUITP404 | Software Engineering Practical | **Skill Enhancement** |
| RJSUITP405 | Computer Graphics and Animation  Practical | **Skill Enhancement** |