Jaya College of Arts and Science, Thiruninravur-602024.

Department of Computer Applications

Year : 2020-2021

Programme Offered :

- **B.C.A**

Programme Objective :

PO 1:	Computational Knowledge: Understand and apply mathematical foundation, computing and domain knowledge for the conceptualization of computing models from defined problems.
PO 2:	Problem Analysis: Ability to identify, critically analyze and formulate complex computing problems using fundamentals of computer science and application domains.
PO 3:	Design / Development of Solutions: Ability to transform complex business scenarios and contemporary issues into problems, investigate, understand and propose integrated solutions using emerging technologies.
PO 4:	Conduct Investigations of Complex Computing Problems: Ability to devise and conduct experiments, interpret data and provide well informed conclusions.
PO 5:	Modern Tool Usage: Ability to select modern computing tools, skills and techniques necessary for innovative software solutions
PO 6:	Professional Ethics: Ability to apply and commit professional ethics and cyber regulations in a global economic environment.
PO 7:	Life-long Learning: Recognize the need for and develop the ability to engage in continuous learning as a Computing professional.
PO 8:	Project Management: Ability to understand management and computing principles with computing knowledge to manage projects in multidisciplinary environments.
PO 9:	Communication Efficacy: Communicate effectively with the computing community as well as society by being able to comprehend effective documentations and presentations.
PO 10:	Societal & Environmental Concern: Ability to recognize economical, environmental, social, health, legal, ethical issues involved in the use of computer technology and other consequential responsibilities relevant to professional practice.
PO 11:	Individual & Team Work: Ability to work as a member or leader in diverse teams in multidisciplinary environment.
PO 12:	Innovation and Entrepreneurship: Identify opportunities, entrepreneurship vision and use of innovative ideas to create value and wealth for the betterment of the individual and society.

Programme Outcome :

Programme Specific Outcomes	
PSO 1:	Understand the basic principles and concepts of Computer applications and integrate the knowledge gained in Computer application domain with practical needs of the society and be an ethically and socially responsible Computer Application Professional
PSO 2:	Explore emerging technologies in diverse areas of Computer Application and inculcate skills for successful career, entrepreneurship and higher studies
PSO 3:	Ability to apply the concepts of Computer and practices via emerging technologies and Software development tools
PSO 4:	Equip themselves to potentially rich & employable field ofcomputer applications.
PSO 5:	Puruse higher studies in the area of Computer Science/Applications.
PSO 6:	Take up self-employment in Indian & global software market.
PSO 7:	Meet the requirements of the Industrial standard

S.NO.	PART	SUBJECT NAME	CREDITS	MAX. MARKS		
SEMESTER I				EXTERNAL MARKS	INTERNAL MARKS	TOTAL
1	I	Tamil/ Other languages – I	3	75	25	100
2	II	English – I	3	75	25	100
3	III	Core I : Problem Solving using Python	4	75	25	100
4	III	Practical I: Problem Solving using Python Lab	2	60	40	100
5	III	Allied I: Mathematics I	5	75	25	100
6	IV	Basic Tamil/Advanced Tamil/Non Major Elective I	2	75	25	100
7	IV	Soft Skill I	3	50	50	100
		Total Credits	**22**			
SEMESTER II			CREDITS	EXTERNAL MARKS	INTERNAL MARKS	TOTAL
8	I	Tamil/ Other languages – II	3	75	25	100
9	II	English – II	3	75	25	100
10	III	Core II :Object Oriented Programming Concepts using C ++	4	75	25	100
11	III	Practical II : C++ programming Lab	3	60	40	100
12	III	Allied II: Mathematics II	5	75	25	100
13	IV	Basic Tamil/Advanced Tamil/Non Major Elective II	2	75	25	100

14	IV	Soft Skill II	3	50	50	100
		Total Credits	**23**			
SEMESTER III			**CREDITS**	**EXTERNAL MARKS**	**INTERNAL MARKS**	**TOTAL**
15	III	Core III: Data Structures	4	75	25	100
16	III	Core IV: Java programming	4	75	25	100
17	III	Core V: Computer Organization	4	75	25	100
18	III	Practical III : Data Structures using Java Lab	3	60	40	100
19	III	Allied III: Financial Accounting	5	75	25	100
20	IV	Soft Skill III	3	50	50	100
21	IV	Environmental Studies	Examination will be held in Semester IV			
		Total Credits	**23**			
SEMESTER IV			**CREDITS**	**EXTERNAL MARKS**	**INTERNAL MARKS**	**TOTAL**
22	III	Core VI: Open Source Technologies	4	75	25	100
23	III	Core VII: Computer Network	4	75	25	100
24	III	Core VIII : E-Commerce technologies	4	75	25	100
25	III	Practical IV : Open Source Technologies Lab	3	60	40	100
26	III	Allied IV: Cost and Management Accounting	5	75	25	100
27	IV	Soft Skill IV	3	50	50	100
28	IV	Environmental Studies	2	75	25	100
		Total Credits	**25**			
SEMESTER V			**CREDITS**	**EXTERNAL MARKS**	**INTERNAL MARKS**	**TOTAL**
29	III	Core IX: Software Engineering	4	75	25	100
30	III	Core X : Operating System	4	75	25	100
31	III	Core XI: Relational Database Management System	4	75	25	100
32	III	Practical V: Operating System Lab	3	60	40	100
33	III	Practical VI : PL/SQL Lab	2	60	40	100
34	III	Elective I	5	75	25	100
35	IV	Value Education	2			
		Total Credits	**24**			
SEMESTER VI			**CREDITS**	**EXTERNAL MARKS**	**INTERNAL MARKS**	**TOTAL**
36	III	Core XII: Web Design and Development	4	75	25	100
37	III	Core XIII: Data Mining	4	75	25	100
38	III	Core XIV: Mobile Application Development	4	75	25	100
39	III	Practical VII: Mobile Application Development Lab	3	60	40	100
40	III	Elective II	5	75	25	100
41	III	Mini Project	5	60	40	100
42	V	Extension Activities	1			
		Total Credits	**26**			
		Total credits (Core, Elective, SBS)	**143**			
Non Major Elective I - I Semester						

Web Application Office Automation HTML
Non Major Elective II – II Semester
Web Application Lab Office Automation Lab HTML Lab
Elective I
Principles of Information Security Resource Management Techniques Multimedia and its Applications
Elective II
Software Project Management IOT and its ApplicationsData Analytics using R

Semester: I

Course Name: PYTHON **Course Code: SE21A**

Course Objective

- Describe the core syntax and semantics of Python programming language.
- Discover the need for working with the strings and functions.
- Illustrate the process of structuring the data using lists, dictionaries, tuples and sets.
- Understand the usage of packages and Dictionaries.

Course OutCome

CO 1:	Understand the principles of Python and acquire skills in programming in python
CO 2:	To develop the emerging applications of relevant field using Python
CO 3:	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.
CO 4:	Able to develop simple turtle graphics programs in Python
CO 5:	Acquire Object Oriented Skills in Python
CO 6:	Educate to explore in the fields of data science, Game development, web development.

Syllabus

UNIT – I

Introduction: The essence of computational problem solving – Limits of computational problem solving-Computer algorithms-Computer Hardware-Computer Software-The process of computational problem solving-Python programming language - Literals - Variables and Identifiers - Operators - Expressions and Data types.

UNIT - II

Control Structures: Boolean Expressions - Selection Control - If Statement- Indentation in Python- Multi-Way Selection -- Iterative Control- While Statement- Infinite loops- Definite vs. Indefinite Loops- Boolean Flags and Indefinite Loops. Lists: List Structures - Lists in Python - Iterating over lists in Python.

UNIT - III

Functions: Program Routines- Defining Functions- More on Functions: Calling Value-Returning Functions-Calling Non-Value-Returning Functions- Parameter Passing - Keyword Arguments in Python - Default Arguments in Python-Variable Scope.

UNIT -I V

Objects and their use: Software Objects - Turtle Graphics – Turtle attributes-Modular Design: Modules - Top-Down Design - Python Modules - Text Files: Opening, reading and writing text files - String Processing - Exception Handling.

UNIT - V

Dictionaries and Sets: Dictionary type in Python - Set Data type. Object Oriented Programming using Python: Encapsulation - Inheritance – Polymorphism. Recursion: Recursive Functions.

Course Name : PYTHON PROGRAMMING LAB Course Code : SE211

Course Objective

- To implement the python programming features in practical applications.
- To write, test, and debug simple Python programs.
- To implement Python programs with conditionals and loops.
- Use functions for structuring Python programs.
- Represent compound data using Python lists, tuples, dictionaries , turtles, Files and modules.

Course OutCome

- Understand the numeric or real life application problems and solve them.
- Apply a solution clearly and accurately in a program using Python.
- Apply the best features available in Python to solve the situational problems.

LIST OF EXERCISES:

1. Program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice.
2. Program to calculate total marks, percentage and grade of a student. Marks obtained in each of the five subjects are to be input by user. Assign grades according to the following criteria:

Grade A: Percentage >=80 Grade B: Percentage >=70 and <80
Grade C: Percentage >=60 and <70 Grade D: Percentage >=40 and <60
Grade E: Percentage <40

3. Program, to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.
4. Program to display the first n terms of Fibonacci series.
5. Program to find factorial of the given number using recursive function.
6. Write a Python program to count the number of even and odd numbers from array of N numbers.
7. Python function that accepts a string and calculate the number of upper case letters and lower case letters.
8. Python program to reverse a given string and check whether the give string is palindrome or not.
9. Write a program to find sum of all items in a dictionary.
10. Write a Python program to construct the following pattern, using a nested loop

1
22
333
4444
55555
666666
7777777
88888888
999999999

11. Read a file content and copy only the contents at odd lines into a new file.
12. Create a Turtle graphics window with specific size.
13. Write a Python program for Towers of Hanoi using recursion
14. Create a menu driven Python program with a dictionary for words and their meanings.
15. Devise a Python program to implement the Hangman Game.

Semester: II

Course Name: Object Oriented Programming Concepts using C ++

Course Code : SU22A

Course Objective

- To inculcate knowledge on Object-oriented programming concepts using C++.
- To gain Knowledge on programming with C++.

Course OutCome

Object Oriented Programming Concepts using C ++	
CO 1:	Compare OOPS with other programming techniques
CO 2:	Implement C++ programs with constructors and destructors
CO 3:	Develop OOP involving polymorphism using operator overloading and method overloading
CO 4:	Implement programs with code reusability using inheritance
CO 5:	Develop Programs with file handling and templates

Syllabus

UNIT - I

Introduction to C++ - key concepts of Object-Oriented Programming –Advantages – Object Oriented Languages – I/O in C++ - C++ Declarations. Control Structures : - Decision Making and Statements : If ..else, jump, goto, break, continue, Switch case statements - Loops in C++ : for, while, do - functions in C++ - inline functions – Function Overloading.

UNIT - II

Classes and Objects: Declaring Objects – Defining Member Functions – Static Member variables and functions – array of objects –friend functions – Overloading member functions – Bit fields and classes – Constructor and destructor with static members.

UNIT- III

Operator Overloading: Overloading unary, binary operators – Overloading Friend functions – type conversion – Inheritance: Types of Inheritance – Single, Multilevel, Multiple, Hierarchal, Hybrid, Multi path inheritance – Virtual base Classes – Abstract Classes.

UNIT - IV

Pointers – Declaration – Pointer to Class , Object – this pointer – Pointers to derived classes and Base classes – Arrays – Characteristics – array of classes – Memory models – new and delete operators – dynamic object – Binding, Polymorphism and Virtual Functions.

UNIT - V

Files – File stream classes – file modes – Sequential Read / Write operations – Binary and ASCII Files – Random Access Operation – Templates – Exception Handling - String – Declaring and Initializing string objects – String Attributes – Miscellaneous functions .

Course Name: C++ PROGRAMMING LAB
Course Code : SU221
Course Objective

- To implement the various object oriented programming concepts using C++

Course Outcome

- To understand the structure and model of the C++ programming language.
- To solve problems in C++ demonstrating Object Oriented Concepts.

LIST OF EXERCISES:

1. Write a C++ program to demonstrate function overloading, Default Arguments and Inline function.
2. Write a C++ program to demonstrate Class and Objects
3. Write a C++ program to demonstrate the concept of Passing Objects to Functions
4. Write a C++ program to demonstrate the Friend Functions.
5. Write a C++ program to demonstrate the concept of Passing Objects to Functions
6. Write a C++ program to demonstrate Constructor and Destructor
7. Write a C++ program to demonstrate Unary Operator Overloading
8. Write a C++ program to demonstrate Binary Operator Overloading
9. Write a C++ program to demonstrate: Single Inheritance , Multilevel Inheritance , Multiple Inheritance , Hierarchical Inheritance , Hybrid Inheritance
10 Write a C++ program to demonstrate Virtual Functions.
11. Write a C++ program to manipulate a Text File.
12. Write a C++ program to perform Sequential I/O Operations on a file.
13. Write a C++ program to find the Biggest Number using Command Line Arguments
14. Write a C++ program to demonstrate Class Template
15. Write a C++ program to demonstrate Function Template.
16. Write a C++ program to demonstrate Exception Handling.

Semester: III

Course Name: DATA STRUCTURES
Course Code : SZ23A
Course Objective

- To understand the concepts of ADTs
- To learn linear data structures-lists, stacks, queues
- To apply Tree and Graph structures
- To understand sorting, searching and hashing

Course OutCome

CO 1:	Implement abstract data types for linear data structures.
CO 2:	Apply the different linear and non linear data structures to problem solutions.
CO 3:	Critically analyze the various sorting algorithms.
CO 4:	To access how the choices of data structure & algorithm methods impact the performance of program.
CO 5:	To Solve problems based upon different data structure & also write programs.
CO 6:	Choose an appropriate data structure for a particular problem.

Syllabus

UNIT - I
Abstract Data Types (ADTs)- List ADT-array-based implementation-linked list implementation-singly linked lists-circular linked lists-doubly-linked lists-applications of lists-Polynomial Manipulation- All operations-Insertion-Deletion-Merge-Traversal.
UNIT - II
Stack ADT-Operations- Applications- Evaluating arithmetic expressions – Conversion of infix to postfix expression-Queue ADT-Operations-Circular Queue- Priority Queue- deQueue-applications of queues.
UNIT - III
Tree ADT-tree traversals-Binary Tree ADT-expression trees-applications of trees-binary search tree ADT-Threaded Binary Trees-AVL Trees- B-Tree- B+ Tree – Heap-Applications of heap.
UNIT - IV
Definition- Representation of Graph- Types of graph-Breadth first traversal – Depth first traversal-Topological sort- Bi-connectivity – Cut vertex- Euler circuits-Applications of graphs.
UNIT - V
Searching- Linear search-Binary search-Sorting-Bubble sort-Selection sort-Insertion sort-Shell sort-Radix sort-Hashing-Hash functions-Separate chaining- Open Addressing-Rehashing-Extendible Hashing.

Course Name: JAVA PROGRAMMING
Course Code : SZ23B
Course Objective

- To understand the concepts of Object Oriented Programming.
- To learn about the control structures, class with attributes and methods used in Java.

Course OutCome

JAVA PROGRAMMING	
CO 1:	Introduces object-oriented design techniques and problem solving.
CO 2:	Knowledge of the structure and model of the Java programming language..
CO 3:	Understand the basic principles of creating Java applications with GUI
CO 4:	Demonstrate use of string and String Buffers, Develop multithreaded programs in Java.
CO 5:	Describe advanced features of Java like exception handling, multi-threading etc.
CO 6:	Ability to work with I/O Streams.

Syllabus

UNIT - I

Introduction to OOPS: Paradigms of Programming Languages – Basic concepts of Object Oriented Programming – Differences between Procedure Oriented Programming and Object Oriented programming - Benefits of OOPs – Application of OOPs. Java: History – Java features – Java Environment – JDK – API. Introduction to Java: Types of java program – Creating and Executing a Java program – Java Tokens- Java Virtual Machine (JVM) – Command Line Arguments –Comments in Java program.

UNIT - II

Elements: Constants – Variables – Data types - Scope of variables – Type casting – Operators: Special operators – Expressions – Evaluation of Expressions. Decision making and branching statements- Decision making and Looping– break – labeled loop – continue Statement. Arrays: One Dimensional Array – Creating an array – Array processing – Multidimensional Array – Vectors – ArrayList – Advantages of Array List over Array Wrapper classes.

UNIT - III

Class and objects: Defining a class – Methods – Creating objects – Accessing class members – Constructors – Method overloading – Static members –Nesting of Methods – this keyword – Command line input. Inheritance: Defining inheritance –types of inheritance– Overriding methods – Final variables and methods – Final classes – Final methods - Abstract methods and classes – Visibility Control- Interfaces: Defining interface – Extending interface - Implementing Interface - Accessing interface variables. Strings: String Array – String Methods – String Buffer Class.

UNIT - IV

Packages: Java API Packages – System Packages – Naming Conventions –Creating & Accessing a Package – Adding Class to a Package – Hiding Classes. Exception Handling: Limitations of Error handling – Advantages of Exception Handling - Types of Errors – Basics of Exception Handling – try blocks – throwing an exception – catching an exception – finally statement. Multithreading: Creating Threads – Life of a Thread – Defining & Running Thread – Thread Methods – Thread Priority – Synchronization –Implementing Runnable interface – Thread Scheduling.

UNIT - V

I/O Streams: File – Streams – Advantages - The stream classes – Byte streams –Character streams. Applets: Introduction – Applet Life cycle – Creating & Executing an Applet –Applet tags in HTML – Parameter tag – Aligning the display - Graphics Class: Drawing and filling lines – Rectangles – Polygon – Circles – Arcs – Line Graphs – Drawing Bar charts AWT Components and Even Handlers: Abstract window tool kit – Event Handlers – Event Listeners – AWT Controls and Event Handling: Labels – Text Component – Action Event – Buttons – Check Boxes – Item Event – Choice– Scrollbars – Layout Managers- Input Events – Menus.

Course Name: COMPUTER ORGANIZATION
Course Code : SZ23C
Course Objective

- To understand the basic organization of computers and the working of each component and CPU
- To bring the programming features of 8085 Microprocessor and know the features of latest microprocessors.
- To understand the principles of Interfacing I/O devices and Direct Memory accesses

Course OutCome

COMPUTER ORGANIZATION	
CO 1:	Describe the major components of a computer system and state their function and purpose
CO 2:	Describe the microstructure of a processor
CO 3:	Demonstrate the ability to program a microprocessor in assembly language.
CO 4:	Classify and describe the operation DMA and peripheral Interfaces
CO 5:	Grasp the different secondary storage devices

Syllabus

UNIT - I
Data representation: Data types – Complements- fixed point and floating point representation other binary codes. Register Transfer and Microoperations: Register transfer language- Register transfer- Bus and Memory transfers – Arithmetic, logic and shift micro operations.
UNIT - II
Central processing unit: General register and stack organizations- instruction formats - Addressing modes- Data transfer and manipulation - program control- RISC - Pipelining - Arithmetic and instruction- RISC pipeline - Vector processing and Array processors.
UNIT - III
Microprocessor Architecture and its Operations - 8085 MPU - 8085 Instruction Set and Classifications. Programming in 8085: Code conversion - BCD to Binary and Binary to BCD conversions - ASCII to BCD and BCD to ASCII conversions - Binary to ASCII and ASCII to Binary conversions.

UNIT - IV
Programming in 8085:BCD Arithmetic - BCD addition and Subtraction - Multibyte Addition and Subtraction - Multiplication and Division. Interrupts: The 8085 Interrupt – 8085 Vectored Interrupts –
UNIT - V
Direct Memory Access(DMA)and 8257 DMA controller - 8255A Programmable Peripheral Interface. Basic features of Advanced Microprocessors - Pentium - I3 , I5 and I7.

Course Name: DATA STRUCTURES USING JAVA LAB
Course Code : SZ231
Course Objective

- To implement linear and non-linear data structures
- To understand the different operations of search trees
- To implement graph traversal algorithms
- To get familiarized to sorting and searching algorithms

Course Outcomes:

- Write functions to implement linear and non-linear data structure operations.
- Suggest appropriate linear and non-linear data structure operations for solving a given problem.
- Analyze various sorting methods.

LIST OF EXERCISES:

1. Write a Java programs to implement the List ADT using arrays and linked lists.

2. Write a Java programs to implement the following using a singly linked list. Stack ADT (b) Queue ADT

3. Write a java program that reads an infix expression, converts the expression to postfix form and then evaluates the postfix expression (use stack ADT).

4. Write a Java program to implement priority queue ADT.

5. Write a Java program to perform the following operations:
(a) Insert an element into a binary search tree.
(b) Delete an element from a binary search tree.
(c) Search for a key element in a binary search tree.
6. Write a Java program to perform the following operations
(a) Insertion into an AVL-tree
(b) Deletion from an AVL-tree
7. Write a Java programs for the implementation of BFS for a given graph.
8. Write a Java programs for the implementation of DFS for a given graph.
9. Write a Java programs for implementing the following searching methods:
(a) Linear search
(b) Binary search.
10. Write a Java programs for implementing the following sorting methods:
(a) Bubble sort
(b) Selection sort
(c) Insertion sort
(d) Radix sort.

Semester IV

Course Name: OPEN SOURCE TECHNOLOGIES
Course Code : SZ24B
Course Objective

To provide a basic idea of Open source technology, their software development process to understand the role and future of open source software in the industry along with the impact of legal, economic and social issues for such software.

Course Outcomes:

OPEN SOURCE TECHNOLOGIES	
CO 1:	To recognize the benefits and features of Open Source Technology and to interpret, contrast and compare open source products among themselves
CO 2:	Understand the difference between open source software and commercial software.
CO 3:	Familiarity with Linux operating system.
CO 4:	Understanding and development of web applications using open source web technologies like Apache, MySql and PHP (LAMP/XAMP)
CO 5:	Install and manage applications.

Syllabus

UNIT- I
Introduction – Why Open Source – Open Source –Principles, Standards Requirements, Successes – Free Software – FOSS – Internet Application Projects

UNIT- II
Open source – Initiatives, Principles, Methodologies, Philosophy, Platform, Freedom, OSSD, Licenses – Copy right, Copy left, Patent, Zero Marginal Technologies, Income generation opportunities, Internalization

UNIT- III
Case Studies – Apache, BSD, Linux, Mozilla (Firefox), Wikipedia, Joomla, GCC, Open Office.

UNIT- IV
Open Source Project –Starting, Maintaining –Open Source – Hardware, Design, Teaching & Media

UNIT- V
Open Source Ethics – Open Vs Closed Source – Government – Ethics – Impact of Open source Technology – Shared Software – Shared Source

Course Name: **COMPUTER NETWORK**

Course Code: SZ24A

Course Objective

- To understand the concept of Computer network
- To impart knowledge about networking and inter networking devices

Course Outcomes:

CO 1:	Analyse different network models
CO 2:	Analyse and compare a number of data link, network and transport layer
CO 3:	Analysing key networking protocols and their hierarchical relationship in the conceptual model like TCP/IP and OSI
CO 4:	Become familiar with fundamental protocols
CO 5:	Demonstrate basic understanding of network principles.

Syllabus

UNIT - I

Introduction – Network Hardware - Software - Reference Models - OSI and TCP/IP Models - Example Networks: Internet, ATM, Ethernet and Wireless LANs - Physical Layer - Theoretical Basis for Data Communication - Guided Transmission Media.

UNIT - II

Wireless Transmission - Communication Satellites - Telephone System: Structure, Local Loop, Trunks and Multiplexing and Switching. Data Link Layer: Design Issues - Error Detection and Correction.

UNIT - III

Elementary Data Link Protocols - Sliding Window Protocols - Data Link Layer in the Internet - Medium Access Layer - Channel Allocation Problem - Multiple Access Protocols - Bluetooth.

UNIT - IV

Network Layer - Design Issues - Routing Algorithms - Congestion Control Algorithms - IP Protocol - IP Addresses - Internet Control Protocols.

UNIT - V

Transport Layer - Services - Connection Management - Addressing, Establishing and Releasing a Connection - Simple Transport Protocol - Internet Transport Protocols (ITP) - Network Security: Cryptography.

Course Name: **E-COMMERCE TECHNOLOGIES**
Course Code : **SZ24C**
Course Objective

- To provide students with an overview and understanding of e-commerce with a specific emphasis on Internet Marketing.
- To explore the major issues associated with e-commerce-security, privacy, intellectual property rights, authentication, encryption, acceptable use policies, and legal liabilities.

Course Objective

E-COMMERCE TECHNOLOGIES	
CO 1:	Obtain a general understanding of basic business management concepts.
CO 2:	Have complete knowledge about basic technical concepts relating to E-Commerce.
CO 3:	Obtain thorough understanding about the security issues, threats and challenges of E-Commerce.
CO 4:	Understand the Internet Architecture and Electronic Payment System.
CO 5:	Evaluate E-commerce models and identify the requirements for starting up and operating E-business sites

Syllabus

UNIT - I

History of E-commerce and Indian Business Context: E-Commerce –Emergence of the Internet –Emergence of the WWW – Advantages of E-Commerce – Transition to E-Commerce in India – The Internet and India – E-transition Challenges for Indian Corporate. Business Models for E-commerce: Business Model – E-business Models Based on the Relationship of Transaction Parties - E-business Models Based on the Relationship of Transaction Types.

UNIT - II

Enabling Technologies of the World Wide Web: World Wide Web – Internet Client-Server Applications – Networks and Internets – Software Agents – Internet Standards and Specifications – ISP. e-Marketing :Traditional Marketing – Identifying Web Presence Goals – Online Marketing – E-advertising – E-branding.

UNIT - III

E-Security: Information system Security – Security on the Internet – E-business Risk Management Issues – Information Security Environment in India. Legal and Ethical Issues : Cybers talking – Privacy is at Risk in the Internet Age – Phishing – Application Fraud – Skimming – Copyright – Internet Gambling – Threats to Children.

UNIT - IV

e-Payment Systems: Main Concerns in Internet Banking – Digital Payment Requirements – Digital Token-based e-payment Systems – Classification of New Payment Systems – Properties of Electronic Cash – Cheque Payment Systems on the Internet – Risk and e-Payment Systems – Designing e-payment Systems – Digital Signature – Online Financial Services in India - Online Stock Trading.

UNIT - V

Information systems for Mobile Commerce: What is Mobile Commerce? – Wireless Applications –Cellular Network – Wireless Spectrum – Technologies for Mobile Commerce – Wireless Technologies –Different Generations in Wireless Communication – Security Issues Pertaining to Cellular Technology. Portals for E-Business: Portals – Human Resource Management – Various HRIS Modules.

Course Name : OPEN SOURCE TECHNOLOGIES LAB

Course Code : SZ241

Course Objective:

- To be aware of the various open source software available for different problem needs
- To be familiar with the usage of the software like installation and configuration

Course Outcomes:

- Students must be able to use appropriate open source tools based on the nature of the problem
- Students should be able to code and compile different open source software

LIST OF EXERCISES:

1. Study and usage of Libre Office Suite – Writer, Calc& Impress
2. Text Processing with PERL
3. Simple Applications using PHP
4. Simple Applications using Python
5. Image editing using GIMP
6. Study and usage of Business Intelligence tools – BIRT, JMagallanes
7. Creation of network diagrams using GraphViz
8. Linux Installation
9. Software Configuration in Linux environment.
10. Version Control System using Git.

Semester V

Course Name: SOFTWARE ENGINEERING
Course Objective

- To introduce the software development life cycles
- To introduce concepts related to structured and objected oriented analysis & design co
- To provide an insight into UML and software testing techniques

Course Outcomes:

- The students should be able to specify software requirements, design the software using tools
- To write test cases using different testing techniques.

Syllabus

UNIT- I
Introduction – Evolution – Software Development projects – Emergence of Software Engineering.
Software Life cycle models – Waterfall model – Rapid Application Development – Agile Model – Spiral Model
UNIT- II
Requirement Analysis and Specification – Gathering and Analysis – SRS – Formal System Specification
UNIT- III
Software Design – Overview – Characteristics – Cohesion & Coupling – Layered design – Approaches
Function Oriented Design – Structured Analysis – DFD – Structured Design – Detailed design
UNIT- IV
Object Modeling using UML – OO concepts – UML – Diagrams – Use case, Class, Interaction, Activity, State Chart – Postscript
UNIT- V
Coding & Testing – coding – Review – Documentation – Testing – Black-box, White-box, Integration, OO Testing, Smoke testin

Course Name : OPERATING SYSTEM
Course Objective:

- To understand the fundamental concepts and role of Operating System.
- To learn the Process Management and Scheduling Algorithms
- To understand the Memory Management policies
- To gain insight on I/O and File management techniques

Course Outcomes:

OPERATING SYSTEM	
CO 1:	Understand the structure and functions of Operating System
CO 2:	Compare the performance of Scheduling Algorithms
CO 3:	Analyze resource management techniques
CO 4:	Identify the features of I/O and File handling methods
CO 5:	The course will cover an introduction on the policies for scheduling, kernel, processes, deadlocks, memory management, synchronization, system calls, and file systems.

Syllabus

UNIT - I
Introduction: Views - Types of System - OS Structure – Operations - Services – Interface- System Calls- System Structure - System Design and Implementation. Process Management: Process - Process Scheduling - Inter-process Communication. CPU Scheduling: CPU Schedulers - Scheduling Criteria - Scheduling Algorithms.
UNIT - II
Process Synchronization: Critical- Section Problem - Synchronization Hardware Semaphores - Classical Problems of Synchronization - Monitors. Deadlocks: Characterization - Methods for Handling Deadlocks - Deadlock Prevention - Avoidance - Detection - Recovery.
UNIT - III
Memory Management: Hardware - Address Binding – Address Space - Dynamic Loading and Linking – Swapping – Contiguous Allocation - Segmentation - Paging – Structure of the Page Table.
UNIT - IV
Virtual Memory Management: Demand Paging - Page Replacement Algorithms - Thrashing. File System: File Concept -. Access Methods - Directory and Disk Structure - Protection - File System Structures - Allocation Methods - Free Space Management.
UNIT - V
I/O Systems: Overview - I/O Hardware - Application I/O Interface - Kernel I/O Subsystem - Transforming 1/0 Requests to Hardware Operations - Performance. System Protection: Goals - Domain - Access matrix. System Security: The Security Problem - Threats – Encryption- User Authentication.

Course Name : RELATIONAL DATABASE MANAGEMENT SYSTEM
Course Objective:

- Gain a good understanding of the architecture and functioning of Database Management Systems
- Understand the use of Structured Query Language (SQL) and its syntax.
- Apply Normalization techniques to normalize a database.
- Understand the need of transaction processing and learn techniques for controlling the consequences of concurrent data access.

Course Outcomes:

RELATIONAL DATABASE MANAGEMENT SYSTEM	
CO 1:	Describe basic concepts of database system
CO 2:	Design a Data model and Schemas in RDBMS
CO 3:	Competent in use of SQL
CO 4:	Analyze functional dependencies for designing robust Database

Syllabus

UNIT - I
Introduction to DBMS– Data and Information - Database – Database Management System – Objectives - Advantages – Components - Architecture. ER Model: Building blocks of ER Diagram – Relationship Degree – Classification – ER diagram to Tables – ISA relationship – Constraints – Aggregation and Composition – Advantages
.
UNIT - II
Relational Model: CODD's Rule- Relational Data Model - Key - Integrity – Relational Algebra Operations – Advantages and limitations – Relational Calculus – Domain Relational Calculus - QBE.
UNIT - III
Structure of Relational Database. Introduction to Relational Database Design - Objectives – Tools – Redundancy and Data Anomaly – Functional Dependency - Normalization – 1NF – 2NF – 3NF – BCNF. Transaction Processing – Database Security.
UNIT - IV
SQL: Commands – Data types – DDL - Selection, Projection, Join and Set Operations – Aggregate Functions – DML – Modification - Truncation - Constraints – Subquery.

UNIT - V
PL/SQL: Structure - Elements – Operators Precedence – Control Structure – Iterative Control - Cursors - Procedure - Function - Packages – Exceptional Handling - Triggers.

Course Name : OPERATING SYSTEM LAB
Course Objective:

- To learn Process management and scheduling.
- To understand the concepts and implementation of memory management policies.
- To understand the various issues in Inter Process Communication.

OUTCOMES:

- Understand the process management policies and scheduling process by CPU.
- Analyze the memory management and its allocation policies.
- To evaluate the requirement for process synchronization.

PROGRAM LIST:

1. Basic I/O programming.To implement CPU Scheduling Algorithms:
2. Shortest Job First Algorithm.
3. First Come First Served Algorithm.
4. Round Robin and Priority Scheduling Algorithms.
5. To implement reader/writer problem using semaphore.
6. To implement Banker's algorithm for Deadlock avoidance. Program for page replacement algorithms:
7. First In First Out Algorithm.
8. Least Recently Used Algorithm.
9. To implement first fit, best fit and worst fit algorithm for memory management.
10. Program for Inter-process Communication.

Course Name : PL/SQL LAB

Course Objective:

- Learn the various DDL and DML commands
- Understand queries in SQL to retrieve information from data base
- Understand PL/SQL statements: Exception Handling, Cursors, and Triggers.
- Develop database applications using front-end and back-end tools.

OUTCOMES:

- Implement the DDL , DML Commands and Constraints
- Create, Update and query on the database.
- Design and Implement simple project with Front End and Back End.

LIST OF EXERCISES

1) DDL commands with constraints.
2) DML Commands with constraints.
3) SQL Queries: Queries, sub queries, Aggregate function
4) PL/SQL : Exceptional Handling
5) PL/SQL : Cursor
6) PL/SQL : Trigger
7) PL/SQL : Packages
8) Design and Develop Application for Library Management
9) Design and Develop Application for Student Mark Sheet Processing
10) Design and Develop Application for Pay Roll Processing

Semester VI

Course Name : WEB DESIGN AND DEVELOPMENT

Course Objective:

- To understand Web based programming and scripting languages.
- To learn the basic web concepts and to create rich internet applications that use most recent client-side programming technologies.
- To learn the basics of HTML, DHTML, XML, CSS, Java Script and AJAX.

Course OutComes:

- Ability to Develop and publish Web pages using Hypertext Markup Language (HTML).
- Ability to optimize page styles and layout with Cascading Style Sheets (CSS).
- Ability to Understand, analyze and apply the role of languages to create a capstone
- Website using client-side web programming languages like HTML, DHTML, CSS, XML, JavaScript, and AJAX.

Syllabus

UNIT I:
HTML: HTML-Introduction-tag basics- page structure-adding comments working with texts, paragraphs and line break. Emphasizing test- heading and horizontal rules-list-font size, face and color-alignment-links-tables-frames

UNIT II:
Forms & Images Using Html: Graphics: Introduction-How to work efficiently with images in web pages, image maps, GIF animation, adding multimedia, data collection with html forms textbox, password, list box, combo box, text area, tools for building web page front page

UNIT III:
XML & DHTML: Cascading style sheet (CSS)-what is CSS-Why we use CSS-adding CSS to your web pages-Grouping styles-extensible markup language (XML). Dynamic HTML: Document object model (DCOM)-Accessing HTML & CSS through DCOM Dynamic content styles & positioning-Event bubbling-data binding.

UNIT IV:
JavaScript : Client side scripting, What is JavaScript, How to develop JavaScript, simple JavaScript, variables, functions, conditions, loops and repetition, Advance script, JavaScript and objects, JavaScript own objects, the DOM and web browser environments, forms and validations

UNIT V:
Ajax: Introduction, advantages &disadvantages, Purpose of it, ajax based web application, alternatives of ajax Java Script & AJAX: Introduction to array-operators, making statements-date & time-mathematics-strings-Event handling-form properties. AJAX. Introduction to jQuery and AngularJS.

Course Name : DATA MINING

Course Objective:

- To learn about data mining Concepts
- To study the different data mining techniques

Course OutComes:

DATA MINING	
CO 1:	To have knowledge in Data mining concepts
CO 2:	To apply Data mining concepts in different fields
CO 3:	Identify the key processes of data mining, data warehousing and knowledge discovery process.
CO 4:	Understand the basic principles and algorithms used in practical data mining and their strengths and weaknesses.
CO 5:	Apply data mining techniques to solve problems in other disciplines in a mathematical way.

Syllabus

UNIT - I

Basic Data Mining Tasks – Data Mining Versus Knowledge Discovery in Data Bases – Data Mining Issues – Data Mining Matrices – Social Implications of Data Mining – Data Mining from Data Base Perspective.

UNIT - II

Data Mining Techniques – a Statistical Perspective on data mining – Similarity Measures – Decision Trees – Neural Networks – Genetic Algorithms.

UNIT - III

Classification: Introduction – Statistical – Based Algorithms – Distance Based Algorithms – Decision.

UNIT - IV

Clustering Tree – Based Algorithms – Neural Network Based Algorithms – Rule Based Algorithms – Combining Techniques: Introduction – Similarity and Distance Measures – Outliers – Hierarchical Algorithms. Partitioned Algorithms.

UNIT - V

Association Rules: Introduction - Large Item Sets – Basic Algorithms – Parallel & Distributed Algorithms – Comparing Approaches – Incremental Rules – Advanced Association Rules Techniques – Measuring the Quality of Rules.

Course Name : MOBILE APPLICATION DEVELOPMENT

Course Objective:

- To make the student understand the basic concepts of mobile application development, be aware of Characteristics of mobile applications, User-interface design, basics of graphics and multimedia.
- To gain knowledge about testing and publishing of Android application

Course OutComes:

MOBILE APPLICATION DEVELOPMENT	
CO 1:	To explain the basics of mobile application development
CO 2:	Develop Android application with User interface, networking and animation.
CO 3:	Use simulator tools to test and publish the application.
CO 4:	Understand the concepts of android operating system and its development tools.
CO 5:	Gain conceptual understanding of Android Activities and GUI Design Concepts
CO 6:	Able to demonstrate their ability to develop software with reasonable complexity in mobile platform.

Syllabus

UNIT - I

Mobile Application Development - Mobile Applications and Device Platforms - Alternatives for Building Mobile Apps -Comparing Native vs. Hybrid Applications -The Mobile Application Development Lifecycle-The Mobile Application Front-End-The Mobile Application Back-End-Key Mobile Application Services-What is Android-Android version history-Obtaining the Required Tools- Launching Your First Android Application-Exploring the IDE-Debugging Your Application-Publishing Your Application

UNIT - II

Understanding Activities-Linking Activities Using Intents-Fragments-Displaying Notifications- Understanding the Components of a Screen-Adapting to Display Orientation-Managing Changes to Screen Orientation-Utilizing the Action Bar-Creating the User Interface Programmatically Listening for UI Notifications

UNIT - III

Using Basic Views-Using Picker Views -Using List Views to Display Long Lists-Understanding Specialized Fragments - Using Image Views to Display Pictures -Using Menus with Views-Using WebView- Saving and Loading User Preferences-Persisting Data to Files-Creating and Using Databases.

UNIT - IV

Sharing Data in Android-Creating Your Own Content Providers -Using the Content Provider- SMS Messaging - Sending Email-Displaying Maps- Getting Location Data- Monitoring a Location.

UNIT - V

Consuming Web Services Using HTTP-Consuming JSON Services- Creating Your Own Services - Binding Activities to Services -Understanding Threading .

Course Name : MOBILE APPLICATION DEVELOPMENT LAB

Course Objective:

- To give overall view of Mobile application development
- Develop and Publish Android applications using Graphical user interface
- Develop and Publish Android application which can use Location and network services

Course Outcomes:

- At the end of the course, the student should be able to:
- Use Emulator tools to design and develop applications

Exercises

1. Develop an application that finds greatest among three numbers using GUI Components
2. Develop an application to display your personal details using GUI Components
3. Develop an application that uses the radio button
4. Develop an application that uses the image button
5. Develop an application that uses Alert Dialog Box
6. Develop an application that uses Layout Managers.
7. Develop an application that uses audio mode (NORMAL, SILENT, VIBRATE)
8. Develop an application that uses to send messages from one mobile to another mobile.
9. Develop an application that uses to send email
10. Develop an application for mobile calls.
11. Develop an application for Student Mark sheet processing
12. Develop an application for Login Page in Database.
13. Develop an application for Google map locator (optional)

Course Name : IOT AND ITS APPLICATIONS

Course Objective:

- To understand the concepts of Internet of Things and the application of IoT.
- To Determine the Market perspective of IoT.
- To Understand the vision of IoT from a global context

Course OutComes:

IOT AND ITS APPLICATIONS	
CO 1:	Use of Devices, Gateways and Data Management in IoT.
CO 2:	Design IoT applications in different domain and be able to analyze their performance
CO 3:	Implement basic IoT applications on embedded platform
CO 4:	Examine the potential business opportunities that IoT can uncover
CO 5:	Explore the relationship between IoT, cloud computing, and big data
CO 6:	Identify how IoT differs from traditional data collection systems.

UNIT - I

IoT & Web Technology, The Internet of Things Today, Time for Convergence, Towards the IoT Universe, Internet of Things Vision, IoT Strategic Research and Innovation Directions, IoT Applications, Future Internet Technologies, Infrastructure, Networks and Communication, Processes, Data Management, Security, Privacy & Trust, Device Level Energy Issues, IoT Related Standardization, Recommendations on Research Topics.

UNIT - II

M2M to IoT – A Basic Perspective– Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven global value chain and global information monopolies. M2M to IoT-An Architectural Overview– Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations.

UNIT - III

IoT Architecture -State of the Art – Introduction, State of the art, Architecture. Reference Model- Introduction, Reference Model and architecture, IoT reference Model, IoT Reference Architecture- Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views.

UNIT - IV

IoT Applications for Value Creations Introduction, IoT applications for industry: Future Factory Concepts, Brownfield IoT, Smart Objects, Smart Applications, Four Aspects in your Business to Master IoT, Value Creation from Big Data and Serialization, IoT for Retailing Industry, IoT For Oil and GasIndustry, Opinions on IoT Application and Value for Industry, Home Management, eHealth.

UNIT - V

Internet of Things Privacy, Security and Governance Introduction, Overview of Governance, Privacy and Security Issues, Contribution from FP7 Projects, Security, Privacy and Trust in IoT-Data-Platforms for Smart Cities, First Steps Towards a Secure Platform, Smartie Approach. Data Aggregation for the IoT in Smart Cities, Security