DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE STRUCTURE - R20

II YEAR I SEMESTER							
S.No	**Subjects**	**L**	**T**	**P**	**C**	**I**	**E**
1	Discrete Mathematical Structures	3	0	0	3	30	70
2	Data Structures	3	0	0	3	30	70
3	Database Management Systems	3	0	0	3	30	70
4	Object Oriented Programming thru Java	3	0	0	3	30	70
5	Computer Organization	3	0	0	3	30	70
6	Data Structures Lab	0	0	3	1.5	30	70
7	Database Management Systems Lab	0	0	3	1.5	30	70
8	Object Oriented Programming thru Java Lab	0	0	3	1.5	30	70
9	Skill Oriented Course – I 1) Animations – 2D Animation 2) Web Application Development Using Full Stack – Frontend Development – Module - I	0	0	4	2		
10	Environmental Science	2	0	0	0		
	Total Credits				**21.5**	**240**	**560**

II Year - I Semester		L	T	P	C
		3	0	0	3
DISCRETE MATHEMATICAL STRUCTURES					

Course Objectives:

- Check the validity of arguments by using basic connective and valid rules of inference.
- Observe various properties of sets and relations.
- Identify different graphs, isomorphism of graphs, paths, cycles and circuits.
- Identify different types of trees.
- To introduce recurrence relations.

Course Outcomes:

1.Ability to apply mathematical logic to solve problems.

2.Understand sets, relations, functions and discrete structures

3.Apply graph theory concepts to modeling problems in Computer Science using graphs.

4.Apply graph theory concepts to modeling problems in Computer Science using trees.

5.Solve different recurrence relations.

UNIT I

Mathematical Logic : Propositional Calculus: Statements and Notations, Connectives, Truth Tables, Tautologies, Equivalence of Formulas, Duality law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof. Predicate calculus: Predicate Logic, Statement Functions, Variables and Quantifiers, Inference theory for predicate calculus.

UNIT II

Set Theory : Introduction, Operations on Binary Sets. Relations: Properties of Binary Relations, Relation Matrix and Digraph, Operations on Relations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering Relations, Hasse Diagrams.

Functions:Bijective Functions, Composition of Functions.

UNIT III

Graph Theory I : Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs:

Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, (Problems and Theorems without proofs).

UNIT IV

Graph Theory II Planar Graphs, Euler's Formula, Graph Colouring and Covering, Chromatic Number, (Problems and Theorems without proofs).

Trees, Directed trees, Binary Trees, Spanning Trees: Properties, Algorithms for Spanning trees and Minimum SpanningTrees.

UNIT V

Recurrence Relations: Generating Functions, Calculating Coefficient of Generating Functions, Recurrence Relations, Formulation as Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots, Solving Inhomogeneous Recurrence Relations.

TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill.
2. Elements of Discrete Mathematics-A Computer Oriented Approach, C. L. Liu and D. P. Mohapatra, 3rdEdition, Tata McGraw Hill.
3. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K. H. Rosen, 7th Edition, Tata McGraw Hill.

REFERENCE BOOKS:

1. Discrete Mathematics for Computer Scientists and Mathematicians, J. L. Mott, A. Kandel, T.P. Baker, 2nd Edition, Prentice Hall of India.
2. Discrete Mathematical Structures, BernandKolman, Robert C. Busby, Sharon Cutler Ross, PHI.
3. Discrete Mathematics, S. K. Chakraborthy and B.K. Sarkar, Oxford, 2011.

II Year - I Semester		L	T	P	C
		3	0	0	3
DATA STRUCTURES					

Course Objectives:

- Understand and apply algorithm analysis for various searching and sorting techniques
- Understand the concept of linked lists and be use it in various applications
- Be able to use Stacks and Queues in various applications
- Understand the concept of Trees & Graphs and perform various operations on it
- Understand the concept of Hashing & different types of Hashing Techniques

Course Outcomes:

1. By the end of the course, the students should be able to:
2. Use various searching and sorting techniques, and analyze the complexity of various algorithms
3. Perform various operations on Linked Lists, and use them in various applications
4. Perform various operations on Stacks and Queues, and use them in various applications
5. Perform various operations on Trees and Graphs, and use them in various applications
6. Understand different types of Hashing Techniques

UNIT I

Algorithms, Performance analysis- time complexity and space complexity, Asymptotic Notation-Big Oh, Omega and Theta notations, Complexity Analysis Examples. Searching and Sorting: Linear and binary search methods. Bubble sort, Insertion sort, Selection Sort, Radix Sort, Comparison of sorting methods.

UNIT II

Data structures-Linear and nonlinear data structures, Linear List, Array representation, Linked representation, singly linked lists -insertion, deletion, search operations, doubly linked lists-insertion, deletion operations, circular Linked lists-insertion, deletion operations, Applications of Linked Lists – Polynomial Representation, Sparse Matrix Representation

UNIT-III

Stacks - Representation of Stacks using arrays and linked lists, Applications of stacks -

Expression evaluation - Infix to Postfix Conversion, Evaluating Postfix Expressions, Reversing the list

Queues – Representation of Queues using arrays and linked lists, Applications of Queues, Circular queue, Double Ended Queue -insertion, and deletion.

UNIT IV

Trees- Terminology, Properties of Binary trees, Binary tree representations, recursive and non-recursive binary tree traversals, Priority Queues, Heaps-Max Heap, Min Heap.

Search trees- Binary search tree, Operations of Binary Search Trees - insertion, deletion and search, balanced search trees, AVL trees - Definition, operations.

UNIT-V

Graphs- Introduction, Definition, Graph Representation, Elementary Graph Operations – Vertex Insertion, Vertex Deletion, Edge Insertion, Edge Deletion etc, Graph Traversals

Hashing: Definition, Hash table, Hash function, Collision, Collision Resolution Techniques-Chaining, and Open Addressing.

TEXT BOOKS:

1. Data structures and Algorithms in Java, R.Lafore, Pearson education
2. Data Structures, Using C, Second Edition, Reema Thareja, OXFORD Higher Education.
3. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.
4. Data structures and Algorithm Analysis in Java, M.A.Weiss, 2nd edition, Addison- Wesley (Pearson Education).

REFERENCE BOOKS:

1. Introduction to Algorithms, Third Edition, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stien.
2. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI

II Year - I Semester		L	T	P	C
		3	0	0	3
DATABASE MANAGEMENT SYSTEMS					

Course Objective:

To learn the principles of systematically designing and using large scale Database Management Systems for various applications.

Course Outcomes:

1. Understand database concepts and the use of data models in describing database
2. Create, maintain and manipulate a relational database using SQL
3. Understand the importance of schema refinement & be able to refine the schema
4. Understand how the DBMS manages the execution of transactions
5. Understand and differentiate various file organizations for the representation of data

UNIT I

Introduction to Database Management System, Data Independence- Relation Systems and Others, Database system architecture, Introduction- The Three Levels of Architecture-The External Level- the Conceptual Level- the Internal Level- Mapping- the Database Administrator, Various Data Models

The ER Model - The Relational Model, Relational Calculus, Introduction to Database Design, Database Design and ER Diagrams-Entities Attributes, and Entity Sets-Relationship and Relationship Sets - Conceptual Design with ER Model

UNIT II

The Relational Model – Basic Concepts, Integrity Constraints Over Relations- Key Constraints – Foreign Key Constraints - Relational Algebra Operations - Selection and Projection- Set Operations, Renaming – Joins- Division

SQL – Various parts of SQL, Basic form of SQL Query, Union, Intersect, and Except, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers

UNIT III

Schema Refinement (Normalization) : Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency(1NF, 2NF and 3 NF), concept of surrogate key, Boyce-codd normal form(BCNF), Lossless join and dependency preserving decomposition, Fourth normal form(4NF).

UNIT IV

Transaction Management and Concurrency Control

Transaction, properties of transactions, Various concurrency control techniques – lock based, timestamp based, lock granularity, lock types, 2PL for ensuring serializability, deadlocks – dealing with deadlocks, Database Recovery management : Log based recovery

UNIT V

Overview of Storages and Indexing, Data on External Storage- File Organization and Indexing – Clustered Indexing – Primary and Secondary Indexes, Index Data Structures, Tree-Based Indexing – B Trees, B+ Trees, Hash-Based Indexing – Basic idea, Comparison of File Organization

TEXT BOOKS:

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGraw Hill 3rd Edition
2. Database System Concepts, Abraham Silberschatz, Henry F. Korth

REFERENCES BOOKS:

1. Fundamentals of Database Systems, Elmasri Navate Pearson Education
2. Introduction to Database Systems, C.J.Date Pearson Education
3. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel

II Year - I Semester		L	T	P	C
		3	0	0	3
OBJECT ORIENTED PROGRAMMING THROUGH JAVA					

Course Objectives:

- To identify Java language components and how they work together in applications
- To learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- To learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
- To understand how to design applications with threads in Java

Course Outcomes:

1. Able to realize the concept of Object Oriented Programming & Java Programming Constructs
2. Able to describe the basic concepts of Java such as, classes, objects, packages, Enumeration and various keywords
3. Develop applications using various types of Inheritance and Interfaces.
4. Able to handle exceptions and perform various input/output operations on strings and files.
5. Write programs using multithreading and interface with databases from Java program.

Unit I

Introductionto OOP: Introduction, Need of OOP, Principles of Object Oriented Languages, Procedural languages vs OOP, Applications of OOP, History of Java, JVM, Java Features, Programming Style, Command Line Arguments, Escape Sequence Comments

Data Types, Variables, Operators and Flow of Control: Variables, Primitive Data types, Constants, Identifiers- Naming Conventions, Keywords, Literals, Operators- Binary, Unary, Ternary, Expressions, Precedence rules and Associativity, Primitive Type Conversion and casting, Flow of Control- Branching, Conditional Loops.

Unit II :

Classes and Objects: Class declaration and Modifiers, Class Members, Declaration of Class Object, Object Creation, Access control for Class Members, Defining methods, Overloaded methods, Recursive methods, Constructor, Constructor overloading, static keyword, this keyword.

Inheritance: Types of Inheritance, Deriving classes using Extends keyword, Method Overloading, super keyword, final keyword, Polymorphism- Abstract classes and methods - Overloading - Overriding - final methods and classes

Unit III

Interface: Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Static methods in interface, functional interfaces.

Packages and Java Library: Defining package, Importing packages and classes into programs, Path and class path, Access control, Java.lang package and its classes, wrapper classes, auto – boxing and auto-unboxing, Java util classes and interfaces.

Unit IV

Exception Handling: Introduction, Exception handling techniques- try…catch, throw, throws, finally block, User defined Exception, checked exception, unchecked exception, custom exception, nested try and catch blocks

Input/Output and String Handling: Files and streams- Byte stream, I/O stream, Character StreamFile Reader and Writer, charArrayReader and Writer, Class String, Methods for Extracting characters from strings, String Methods, String Buffer, Class String Buffer.

Unit V

Mutli- Threading: Introduction, Need for Multiple threads, Mulithreaded Programming, Thread Class, Main thread, Creation of new thread, thread states, thread priority.

Java Database Connectivity: Introduction, JDBC Architecture, Environment Setup, JDBC Database Connections, Resultset Interface, Creating JDBC Applications

TEXT BOOKS:

1. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
2. The complete Reference Java, 8th edition, Herbert Schildt, TMH
3. Cay S. Horstmann, Gary cornell, ―Core Java Volume –I Fundamentals, 9th Edition, Prentice Hall,2013.

REFERENCE BOOKS:

1. Introduction to java programming, 7th edition by Y Daniel Liang, Pearson
2. Murach's Java Programming, Joel Murach

E-Resources:

1) https://nptel.ac.in/courses/106/105/106105191/

2) https://www.w3schools.com/java/java_data_types.asp

II Year - I Semester		L	T	P	C
		3	0	0	3
COMPUTER ORGANIZATION					

OBJECTIVE:

- Understand the architecture of a modern computer with its various processing units.

OUTCOMES:

1. Students can understand the architecture of modern computers.
2. Understanding of different instruction types.
3. Students can calculate the effective address of an operand by addressing modes
4. They can understand how computers store positive and negative numbers.
5. Understanding of how a computer performs arithmetic operation of positive and negative numbers.

UNIT I

Basic Structure Of Computers: Functional unit, Basic Operational concepts, Bus structures, System Software, Performance, The history of computer development.

UNIT II

Machine Instruction and Programs: Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Basic Instruction Types, Addressing Modes, Basic Input/output Operations, The role of Stacks and Queues in computer programming equations. Component of Instructions: Logic Instructions, shift and Rotate Instructions, Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations

UNIT III

Input / Output Organization: Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access, Buses: Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface: Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB)

UNIT IV

The Memory Systems: Basic memory circuits, Memory System Consideration, ReadOnly Memory: ROM, PROM, EPROM, EEPROM, Flash Memory, Cache Memories: Mapping Functions, INTERLEAVING Secondary Storage: Magnetic Hard Disks, Optical Disks,

UNIT V

Processing Unit: Fundamental Concepts: Register Transfers, Performing An Arithmetic Or Logic Operation, Fetching A Word From Memory, Execution of Complete Instruction, Hardwired Control, Micro programmed Control: Microinstructions, Microprogram Sequencing, Wide Branch Addressing Microinstructions with next –Address Field

TEXT BOOKS:

1. Computer Organization, Carl Hamacher, Zvonks Vranesic, Safea Zaky, 5th Edition, McGraw Hill.
2. Computer Architecture and Organization, John P. Hayes, 3rd Edition, McGraw Hill.

REFERENCE BOOKS:

1. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI
2. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson
3. Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int.
4. “Computer Organization and Design: The Hardware/Software Interface” by David A. Patterson and John L. Hennessy.

II Year - I Semester		L	T	P	C
		0	0	3	1.5
DATA STRUCTURES LAB					

Course Objectives: The objective of this lab is to demonstrate the different data structures implementation.

Course Outcomes:

1. By the end of this lab the student is able to
2. Use various searching and sorting algorithms.
3. Use basic data structures such as arrays and linked list.
4. Implement various data structures like stacks, queues, trees & graphs, and use them for various applications

List of Experiments:

Exercise -1 (Searching)

a) Write a program that use both recursive and non recursive functions to perform Linear search for a Key value in a given list.

b) Write a program that uses both recursive and non-recursive functions to perform Binary search for a Key value in a given list.

Exercise -2 (Sorting-I) Write programs to implement various sorting techniques like Bubble sort, Selection sort, Insertion sort, Radix sort

Exercise -3(Singly Linked List)

a) Write a program that uses functions to create a singly linked list

b) Write a program that uses functions to perform insertion operation on a singly linked list

c) Write a program that uses functions to perform deletion operation on a singly linked list

d) Write a program to reverse elements of a single linked list.

Exercise -4 (Stack)

a) Write a program that implement stack (its operations) using arrays

b) Write a program that implement stack (its operations) using Linked list

c) Write a program that uses Stack operations to evaluate postfix expression

Exercise -5 (Queue)

a) Write a program that implements Queue (its operations) using arrays.

b) Write a program that implement Queue (its operations) using linked lists

Exercise -6 (Binary Tree) Write a recursive function for traversing a binary tree in preorder, inorder and postorder.

Exercise -7 (Binary Search Tree)

a) Write a program to Create a BST

b) Write a program to insert a node into a BST.

c) Write a program to delete a node from a BST.

Exercise – 8 (Graphs) – Represent graphs using adjacency matrix and adjacency list

II Year - I Semester		L	T	P	C
		0	0	3	1.5
DATABASE MANAGEMENT SYSTEMS LAB					

Course Objectives:

The objective of this lab is to teach the students how to store and retrieve data from database using query languages. In addition, the students should be able to enforce various integrity constraints on the database data.

List of Experiments:

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSECT, Constraints.
3. Queries using Aggregate functions, GROUP BY, HAVING and Creation and Dropping of Views.
4. Queries using Conversion functions, String functions, Date functions
5. i) Creation of simple PL/SQL program which includes declaration section, executable section and exception –handling

   ii) Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, user defined Exceptions, RAISE-APPLICATION ERROR.
8. Program development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10. Program development using creation of package specification, package bodies, private objects, package variables and cursors and calling stored packages.

11. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
12. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers

II Year - I Semester		L	T	P	C
		0	0	3	1.5
OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB					

Course Objectives:

- Practice Programming in the Java
- Gain knowledge of object oriented paradigm in the java programming language
- Learn use of java in a variety of technologies and on different platforms.

Course Outcomes:

1. Apply the basic features of JAVA such as Control statements, Arrays, Classes, Inheritance, Interface and Packages in solving a problem
2. Apply appropriate IO stream and collection framework for solving real time problem
3. Determine Class, Objects, Methods, Exception and Polymorphism.
4. Illustrating Simple Inheritance, multi-level Inheritance , Exception handling mechanism.
5. Construct Threads and Implement Packages.

Exercise - 1 (Basics)

A. Write a JAVA program to display default value of all primitive data type of JAVA
B. Write a java program that displays the roots of a quadratic equation $ax^2 +bx=0$. Calculate the discriminant D and basing on the value of D, describe the nature of the root.
C. Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

Exercise - 2 (Operations, Expressions, Control-flow, Strings)

A. Write the programs using the concept of operators, nested loops, recursion, arrays, String and StringBuffer class.

Exercise - 3 (Class, Objects)

A. Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside the main method.
B. Write a JAVA program to implement constructor.

Exercise - 4 (Methods)

A. Write a JAVA program to implement constructor overloading.
B. Write a JAVA program implementing method overloading.

Exercise - 5 (Inheritance)

A. Write a JAVA program to implement Single Inheritance
B. Write a JAVA program to implement multilevel Inheritance
C. Write a java program for abstract class to find areas of different shapes
D. Write a JAVA program that uses "super" keyword.
E. Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?

Exercise - 6 (Exception)

A. Write a JAVA program that describes exception handling mechanism
B. Write a JAVA program Illustrating Multiple catch clauses

Exercise – 7 (Runtime Polymorphism)

A. Write a JAVA program that implements Runtime polymorphism
B. Write a Case study on run time polymorphism, inheritance that implements in above problem

Exercise – 8 (User defined Exception)

A. Write a JAVA program for creation of Illustrating throw
B. Write a JAVA program for creation of Illustrating finally
C. Write a JAVA program for creation of Java Built-in Exceptions
D. Write a JAVA program for creation of User Defined Exception

Exercise – 9 (Threads)

A. Write a JAVA program that creates threads by extending Thread class .First thread display "Good
B. Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and the third display "Welcome" every 3 seconds ,(Repeat the same by implementing Runnable)
C. Write a program illustrating isAlive and join ()
D. Write a Program illustrating Daemon Threads.
E. Write a JAVA program Producer Consumer Problem

Exercise – 10 (Packages)

A. Write a JAVA program illustrate class path
B. Write a case study on including in classpath in your os environment of your package.

C. Write a JAVA program that import and use the defined your package in the previous Problem

Exercise - 11 (I/O & JDBC)

A. Write a program that uses the I/O package for reading and writing a text file.
B. Write a program that uses JDBC API for interacting with the database.

II Year - I Semester		L	T	P	C
		0	0	4	2
SKILL ORIENTED COURSE – I **ANIMATIONS – 2D ANIMATION**					

Course Objectives:

This Course will enable students to learn various aspects of animation using a variety of 2-D Software and to implement advance principles of traditional animation in Adobe animate to create high Quality animation for production.

Course Outcomes:

1. At the end of the course the student will be able to:
2. learn various tools of digital 2-D animation.
3. understand the production pipeline to create 2-D animation.
4. analyze special effects in animation to bring interest and awe in the scenes and backgrounds.
5. apply the tools to create 2D animation for films and videos.

Perform Experiments related to the following concepts:

2D GRAPHIC DESIGN

Adobe Photoshop:

1. Create your visiting card
2. Create Title for any forthcoming film
3. Digital Matte Paint
4. Convert Black and White to Color
5. Convert Day mode to Night mode
6. Design Image manipulation
7. Smooth skin and remove blemishes & scars
8. Create a 3D pop-out effect
9. Create Textures
10. Timeline Animation

Adobe Illustrator:

1. Advertisement

2. Digital Illustrations
3. Brochure
4. Packet Design(Toothpaste packet, Soap cover, any Food product)
5. Danglers for display
6. Menu cards
7. Calendar Design
8. Tracing image
9. Vehicle Design
10. **Festival**

Adobe Indesign:

1. Magazine A4 Size
2. Newspaper layout design & advertisements – Fine arts
3. Special Supplement
4. Different categories of Books
5. Info-graphics
6. Caricatures

Corel DRAW:

1. Create a paper ad for advertising of any commercial agency
2. Package Design
3. Corporate ID
4. Exhibition Layout
5. Oblers

2D ANIMATION

1. Creating Web Banners in Adobe Flash
2. Creating a Logo Animation in Adobe Flash
3. Creating Frame by Frame animation
4. Draw Cartoon Animation using reference.
5. Create Lip Sink to Characters
6. Using filters & Special effects
7. Create a scene by using Mask layers animation

E-Learning Lab:

1. Student Application form

2. Video Controlling
3. Audio Controlling
4. Start Drag and Stop Drag Actions
5. Interactive Keyboard Controls using Flash Action Script.
6. Interactive Flash Game.
7. **Creating Character Animation in After Effects**

II Year - I Semester		L	T	P	C
		0	0	4	2

SKILL ORIENTED COURSE – I

WEB APPLICATION DEVELOPMENT USING FULL STACK – MODULE - I

Course Objectives:

The objective of this lab is to provide understanding about the core concepts of frontend programming for web application

Course Outcomes:

1. By the end of this lab the student is able to
2. Analyze a web page and identify its elements and attributes.
3. Demonstrate the important HTML tags for designing static pages and separate design from content using Cascading Style sheet
4. Implement MVC and responsive design to scale well across PC, tablet and Mobile Phone
5. Create web pages using HTML and Cascading Style Sheets.

Perform experiments related to the following concepts:

A) HTML

1) Introduction to HTML
2) Browsers and HTML
3) Editor's Offline and Online
4) Tags, Attribute and Elements
5) Doctype Element
6) Comments
7) Headings, Paragraphs, and Formatting Text
8) Lists and Links
9) Images and Tables

B) CSS

1) Introduction CSS
2) Applying CSS to HTML
3) Selectors, Properties and Values

4) CSS Colors and Backgrounds

5) CSS Box Model

6) CSS Margins, Padding, and Borders

7) CSS Text and Font Properties

8) CSS General Topics

II Year - I Semester		L	T	P	C
		2	0	0	0
ENVIRONMENTAL SCIENCE					

Objectives:

To make the student to get awareness on environment, to understand the important of protecting natural recourses, ecosystems for futures generations and pollution causes due to the day to day activates of human life to save Earth from the inventions by the engineers.

Course Outcomes:

Students will be able to

1. Articulate the basic structure, functions, and processes of key social systems affecting the Environment.
2. Explain how Natural Resources should be used.
3. Identify the threats to biodiversity.
4. Understand causes, effects and control measures of Environmental pollution.
5. Gain knowledge about Watershed management and Environmental ethics.Gain a rigorous foundation in various scientific disciplines as they apply to environmental science, such as ecology, evolutionary biology, hydrology, and human behavior.

UNIT I

Multidisciplinary nature of Environmental Science and Ecosystems Definition, Scope and Importance – Sustainability: Need for public awareness-Human population and Environment.

Ecosystems: Concept of an ecosystem. - Structure and function of an Ecosystem. -Types of Ecosystem-Forest, Grassland, Desert and Aquatic Ecosystems– Food chains, food webs and ecological pyramids.

UNIT II

Natural Resources Forest resources: Use and over – exploitation, deforestation – Timber extraction – Mining, dams and other effects on forest and tribal people.

Water resources: Conflicts over water, Dams – benefits and problems.

Mineral resources: Use and exploitation, Environmental effects of extracting and using mineral resources.

Energy resources: Growing energy needs, renewable and non-renewable energy sources.

Food resources: World food problems. Land resources: Wasteland reclamation.

Role of an individual in conservation of natural resources.

UNIT III

Biodiversity and its conservation

Definition, Genetic, species and ecosystem diversity- classification - Value of biodiversity: Consumptive use, Productive use, Social use. Biodiversity at national and local levels.Hot-spots of biodiversity - Threats to biodiversity - Endangered and endemic species of India – Conservation of biodiversity.

UNIT IV

Environmental Pollution Definition, Cause, effects and control measures of Air pollution, Water pollution, Soil pollution, Noise pollution, Nuclear hazards. Role of an individual in prevention of pollution.Pollution case studies.

Solid Waste Management: Sources, effects and control measures of urban and industrial solid wastes. Bio medical and e-waste management.Global Environmental Challenges: Global warming and climate change-Acid rains, Ozone layer depletion.

UNIT V

Social Issues and Environmental Management

Urban problems related to energy -Water conservation, Rain water harvesting-Resettlement and rehabilitation of people. Environmental Protection Act –Air Act –Water Act - Wildlife Protection Act -Forest Conservation Act-Public awareness.

International protocols: Stockholm and Rio Summit, Kyoto protocol and Montreal Protocol.

Impact Assessment and its significance various stages of EIA, Environmental audit, Ecotourism. The student should Visit an Industry / Ecosystem.

Text Books

1. A Textbook of Environmental Studies, Shashi Chawla, TMH, NewDelhi.
2. Textbook of Environmental Studies for Undergraduate Courses by ErachBharucha for University Grants Commission.
3. Environmental Studies, R. Rajagopalan, 2nd Edition, 2011, Oxford UniversityPress.

References

1. Environmental Studies, K. V. S. G. Murali Krishna, VGS Publishers,Vijayawada.
2. Text Book of Environmental Studies, Deekshita Dave & P. Udaya Bhaskar,

CengageLearning.

3. Textbook of Environmental Science and Technology – Dr. Anji Reddy, BS Publications.
4. Environment Studies, Anubha Kaushik, C P Kaushik, New Age International Publishers,2014.
5. Environmental Studies, P. N. Palanisamy, P. Manikandan, A. Geetha, and K. Manjula Rani; Pearson Education,Chennai.
6. Environmental Studies, Benny Joseph, Tata McGraw Hill Co, NewDelhi.