Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal

Branch- Common to All Discipline

ES301	Energy & Environmental Engineering	3L-1T-0P	4 Credits

The objective of this Course is to provide *an introduction to energy systems and renewable energy resources, with a scientific examination of the energy field and an emphasis on alternative energy sources and their technology and application.*

Module 1: Introduction to Energy Science:
Introduction to energy systems and resources; Introduction to Energy, sustainability & the environment; Overview of energy systems, sources, transformations, efficiency, and storage; Fossil fuels (coal, oil, oil-bearing shale and sands, coal gasification) - past, present & future, Remedies & alternatives for fossil fuels - biomass, wind, solar, nuclear, wave, tidal and hydrogen; Sustainability and environmental trade-offs of different energy systems; possibilities for energy storage or regeneration (Ex. Pumped storage hydro power projects, superconductor-based energy storages, high efficiency batteries)

Module2: Ecosystems
- Concept of an ecosystem; Structure and function of an ecosystem; Producers, consumers and decomposers; Energy flow in the ecosystem; Ecological succession; Food chains, food webs and ecological pyramids; Introduction, types, characteristic features, structure and function of the following ecosystem (a.)Forest ecosystem (b) Grassland ecosystem (c) Desert ecosystem (d) Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Module 3: Biodiversity and its conservation
- Introduction – Definition: genetic, species and ecosystem diversity; Bio-geographical classification of India; Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values; Biodiversity at global, National and local levels; India as a mega-diversity nation; Hot-sports of biodiversity; Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts; Endangered and endemic species of India; Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

Module 4: Environmental Pollution

- Definition, Cause, effects and control measures of Air pollution, Water pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclear hazards; Solid waste Management: Causes, effects and control measures of urban and industrial wastes; Role of an individual in prevention of pollution; Pollution case studies; Disaster management: floods, earthquake, cyclone and landslides.

Module 5: Social Issues and the Environment

- From Unsustainable to Sustainable development; Urban problems related to energy; Water conservation, rain water harvesting, watershed management; Resettlement and rehabilitation of people; its problems and concerns. Case Studies
Environmental ethics: Issues and possible solutions. Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies Wasteland reclamation; Consumerism and waste products; Environment Protection Act; Air (Prevention and Control of Pollution) Act; Water (Prevention and control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act;   Issues involved in enforcement of environmental legislation; Public awareness.

Module 6: Field work

- Visit to a local area to document environmental assets-river/forest/grassland/hill/mountain
- Visit to a local polluted site-Urban/Rural/Industrial/Agricultural
- Study of common plants, insects, birds.
- Study of simple ecosystems-pond, river, hill slopes, etc.

REFERENCE

1.   Brunner R.C., 1989, Hazardous Waste Incineration, McGraw Hill Inc.
2.   Clark R.S., Marine Pollution, Clanderson Press Oxford (TB).
3.   Cunningham, W.P. Cooper, T.H. Gorhani, E & Hepworth, M.T. 2001, Environmental Encyclopedia, Jaico Publ. House, Mumabai,
4.   De A.K., Environmental Chemistry, Wiley Eastern Ltd.
5.   Trivedi R.K., Handbook of Environmental Laws, Rules Guidelines, Compliances and Standards', Vol I and II, Enviro Media (R)
6.   Boyle, Godfrey, Bob Everett, and Janet Ramage (Eds.) (2004), Energy Systems and Sustainability: Power for a Sustainable Future. Oxford University Press.
7.   Schaeffer, John (2007), Real Goods Solar Living Sourcebook: The Complete Guide to Renewable Energy Technologies and Sustainable Living, Gaiam

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Information Technology, III-Semester

IT302 Discrete Structure

Course objectives

The main objectives of this course are:

1. To introduce students with sets, relations, functions, graph, and probability.
2. To enable students to perform set operation and solve logical reasoning and verify the correctness of logical statement.
3. To apply the properties of relations and find partially ordered set and lattices.

Unit I-Set Theory, Relation, Function, Theorem Proving Techniques : Set Theory: Definition of sets, countable and uncountable sets, Venn Diagrams, proofs of some general identities on sets Relation: Definition, types of relation, composition of relations, Pictorial representation of relation, Equivalence relation, Partial ordering relation, Job Scheduling problem Function: Definition, type of functions, one to one, into and onto function, inverse function, composition of functions, recursively defined functions, pigeonhole principle. Theorem proving Techniques: Mathematical induction, Proof by contradiction.

Unit II- Algebraic Structures: Definition, Properties, types: Semi Groups, Monoid, Groups, Abelian group, properties of groups, Subgroup, cyclic groups, Normal subgroup, Homomorphism and isomorphism of Groups, example and standard results, Rings and Fields: definition and standard results.

Unit III- Propositional Logic: Proposition, First order logic, Basic logical operation, truth tables, tautologies, Contradictions, Algebra of Proposition, logical implications, logical equivalence, predicates, Normal Forms, Universal and existential quantifiers. Introduction to finite state machine Finite state machines as models of physical system equivalence machines, Finite state machines as language recognizers

Unit IV- Graph Theory: Introduction and basic terminology of graphs, Planer graphs, Multigraphs and weighted graphs, Isomorphic graphs, Paths, Cycles and connectivity, Shortest path in weighted graph, Introduction to Eulerian paths and circuits, Hamiltonian paths and circuits, Graph coloring, chromatic number, Isomorphism and Homomorphism of graphs.

Unit V- Posets, Hasse Diagram and Lattices: Introduction, ordered set, Hasse diagram of partially, ordered set, isomorphic ordered set, well ordered set, properties of Lattices, bounded and complemented lattices. Combinatorics: Introduction, Permutation and combination, Binomial Theorem, Recurrence Relation and Generating Function: Introduction to Recurrence Relation and Recursive algorithms , Linear recurrence relations with constant coefficients, Homogeneous solutions, Particular solutions, Total solutions , Generating functions , Solution by method of generating functions.

Course Outcomes

On completion of the course;

1. Students will be able to understand the notion of mathematical thinking, and algorithmic thinking and be able to apply them in problem solving such as formal specification, verification, and basic concepts of set theory.
2. Students understand the basic principle of Boolean algebra, logic and set theory.
3. Be able to construct simple mathematical proof and possess the ability to verify them.

Reference Books:

1. C.L.Liu" Elements of Discrere Mathematics" TMH.
2. Lipschutz, "Discrete mathematics (Schaum)",TMH.
3. U.S Gupta " Discrete Mathematical Structures" Pearson.
4. S. Santha," Discrete Mathematics with Combinatorics and graph theory", Cengage Learning.
5. Dr.Sukhendu. Dey " Graph Theory With Applications" Shroff Publishers

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Information Technology, III-Semester

IT303 Data Structure

Course objectives

The main objectives of this course are:

1. To introduce the concepts of linear, non-linear data structures , the operations performed on them and the applications of various data structures.
2. To introduce various algorithms of searching and sorting.
3. To understand the basic concepts of stacks, queues, linked lists, trees and graphs
4. To enable students to write algorithms for solving various problems using data structures.

Unit 1: Introduction Data, data type, data object. Types of data structure – primitive &n non-primitive , linear & non-linear. Operations on data structures – traversing, searching , inserting , deleting. Complexity analysis – worst case, best case, average case. Time – space trade off , algorithm efficiency, asymptotic notations – big oh , omega , theta.

Unit 2: Arrays & Structure Introduction , declaration of arrays , operations on arrays – inserting , deleting , merging of two arrays , 1 dimensional & 2 dimensional arrays, row & column major representation , address calculation in array , storing values in arrays , evaluation of polynomial – addition & representation. Searching & sorting – Introduction , sequential search, binary search , Fibonacci search , indexed sequential search, hashed search. Types of sorting with general concepts – bubble , heap , insertion , selection , quick , heap , shell , bucket , radix and merge sort.

Unit 3: Stacks & Queues Basic concept of stacks & queues, array representation of stacks, operation on stacks – push , pop , create , getTop , empty , linked representation of stack , multiple stack. Application of stack – Conversion: infix , prefix , postfix and evaluation of arithmetic expression. Linked representation of queue, operations on queue – insertion & deletion. Types of queue with functions – circular , deque , priority queue. Applications of queues – job scheduling , Josephus problem.

Unit 4: Linked List Introduction – basic terminology , memory allocation & deallocation for linked list. Linked list variants – head pointer , head node , types linked list – linear & circular linked list. Doubly linked list , creation of doubly list, deletion of node from doubly linked list, insertion of a node from doubly linked list, traversal of doubly linked list. Circular linked list – singly circular linked list , circular linked list with header node , doubly circular linked list. Applications of linked list – polynomial representation & garbage collection.

Unit 5: Trees Basic terminology – general tree , representation of general tree, types of trees, binary tree- realization and properties , traversal in binary trees – inorder , preorder , postorder , applications of trees. Graph- Basic Terminologies and representations, Graph search and traversal algorithms.

Course Outcomes

On completion of the course:

1. For a given search problem (linear search and binary search) student will be able to implement it.
2. For a given problem of stacks, queues and link lists, students will be able to implement it and analyze the same to determine the time and computation complexity
3. Students will be able to write an algorithm for selection sort, insertion sort, quick sort, merge sort, heap sort, bubble sort and compare their performance
4. Students will be able to implement tree, graph search and traversal algorithms

References :

1. Varsha H. Patil "Data Structure Using C++" Oxford.
2. Rajesh K. Shukla "Data Structures Using C & C++" Wiley India.
3. Reema Thareja " Data Structure Using C " Oxford.
4. D. S Malik "Data Structure Using C++ " Second Edition Cengage.
5. Kushwaha and Mishra "Data Structure: A programming Approach with C", PHI Learning.
6. A. K Sharma "Data Structure Using C" Pearson.
7. Ellis Horowitz, Sartaj Sahni, "Fundamentals of Data Structures", Computer Science Press

List of Experiments

1. Write a program to search an element in the array using Linear and Binary Search.
2. Write a program to perform the following operation in Matrix:

   1. Addition 2. Subtraction 3. Multiplication 4. Transpose

3. Write a program to perform the following operation on strings using string functions:

   1. Addition 2. Copying 3. Reverse 4. Length of String

4. Write program for implementing the following sorting methods to arrange a list of integers in ascending order:

   a) Quick sort b) Selection sort c) Insertion sort d) Merge sort

5. Write a program that uses stack operations to convert a given infix expression into its postfix equivalent.
6. Write a program to merge two sorted array into one sorted array.
7. Write a program to implement stack using array and linked list.
8. Write a program to implement queue and circular queue using array.
9. Write a program to insert an element in the beginning and end of singly linked list.
10. Write a program to insert an element at any position in singly and doubly linked list.
11. Insert and delete a node at any position in doubly linked list.
12. Write a program of Tower of Hanoi.
13. Write a program that uses functions to perform the following:

   a) Create a binary search tree of integers.

   b) Traverse the above Binary search tree non recursively in in order.

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Information Technology, III-Semester

IT304 Object Oriented Programming & Methodology

Course Objectives
1. The objective of this course is to understand the advantage of object oriented programming over procedure oriented programming.
2. To help students to understand the key features of Object Oriented Programming and Methodology like objects, methods, instance, message passing, encapsulation, polymorphism, data hiding, abstract data and inheritance.
3. To develop understanding of pointers and memory management.
4. To be able to develop understanding of file input/output and templates

Unit I- Introduction: Object oriented programming, Introduction, Application, characteristics, difference between object oriented and procedure programming, Comparison of C and C++, Cout, Cin, Data Type, Type Conversion, Control Statement, Loops, Arrays and string arrays fundamentals, Function, Returning values from functions, Reference arguments, Overloaded function, Inline function, Default arguments, Returning by reference.

Unit II- Object and Classes: Implementation of class and object in C++, access modifiers, object as data
type, constructor, destructor, Object as function arguments, default copy constructor, parameterized constructor, returning object from function, Structures and classes, Classes objects and memory, static class data, Arrays of object, Arrays as class Member Data, The standard C++ String class, Run time and Compile time polymorphism.

Unit III- Operator overloading and Inheritance: Overloading unary operators, Overloading binary operators, data conversion, pitfalls of operators overloading, Concept of inheritance, Derived class and base class, access modifiers, types of inheritance, Derived class constructors, member function, public and private inheritance.

Unit IV- Pointer and Virtual Function: Addresses and pointers, the address-of operator & pointer and arrays, Pointer and Function pointer, Memory management: New and Delete, pointers to objects, debugging pointers, Virtual Function, friend function, Static function, friend class, Assignment and copy initialization, this pointer, dynamic type information.

Unit V-Streams and Files: Streams classes, Stream Errors, Disk File I/O with streams, file pointers, error handling in file I/O with member function, overloading the extraction and insertion operators, memory as a stream object, command line arguments, printer output, Function templates, Class templates Exceptions, Containers, exception handling.

Course Outcomes

On the completion of this course students will be able to:

1. Recognize attributes and methods for given objects.
2. Define data types and also deal with operations applied for data structures.
3. Implement algorithms and complex problems.

Reference Books:

1. E. Balaguruswami, "Object Oriented Programming in C++", TMH.
2. Robert Lafore, "Object Oriented Programming in C++", Pearson.
3. M.T. Somashekare, D.S. Guru, " Object-Oriented Programming with C++", PHI.
4. Herbert Shildt, "The Complete Reference C++", Tata McGraw Hill publication.

List of Experiments:

1. Write a program to find out the largest number using function.
2. Write a program to find the area of circle, rectangle and triangle using function overloading.
3. Write a program to implement complex numbers using operator overloading and type conversion.
4. Write a program using class and object to print bio-data of the students.
5. Write a program which defines a class with constructor and destructor which will count number of object created and destroyed.
6. Write a program to implement single and multiple inheritances taking student as the sample base class.
7. Write a program to add two private data members using friend function.
8. Write a program using dynamic memory allocation to perform 2x2 matrix addition and subtraction.
9. Write a program to create a stack using virtual function.
10. Write a program that store five student records in a file.
11. Write a program to get IP address of the system.
12. Write a program to shutdown the system on windows operating system.

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Information Technology, III-Semester

IT305 Digital Circuits and Systems

Course Objectives
1 Understand working of logic gates.
2 To design and implement combinational and sequential logic circuits
3 Understand the process of analog to digital and digital to analog conversion
4 To understand various logic families

Unit I- Number systems and logic gates: Decimal, Binary, Octal, Hexadecimal number systems and radix conversion. Codes- BCD, excess 3, gray, ASCII. Boolean algebra- Theorems and properties, Boolean functions, canonical and standard forms, De Morgans theorem, digital logic gates, Karnaugh maps.

Unit II- Combinational circuits: Introduction to combinational circuits, multilevel NAND, NOR implementation. Designing binary Adders and Subtractors. Decoder, Encoder, Multiplexer, Demultiplexer circuits.

Unit III- Sequential circuits: Introduction to Sequential circuits, flip-flops, RS, D, T, JK, M/S JK-flipflops, truth tables, excitation tables and characteristic equations, clocked and edge triggered flipflops, Registers- Definition, serial, parallel, shift left/right registers, Johnson counter, asynchronous and synchronous counters.

Unit IV- Digital logic families: Bipolar and unipolar logic families, Digital IC specifications, RTL, DTL, All types of TTL circuits, ECL, IIL, PMOS, NMOS & CMOS Logic.

Unit V- Clocks and timing circuits: Bistable, Monostable & Astable multivibrator, Schmitt trigger circuit, Introduction of Analog to Digital & Digital to Analog converters, Display devices, 7 and 16 segment LED display, LCD.

Course Outcomes
 On the completion of this course
1 Students will be able to perform number base conversions, use Boolean logic to create digital circuits.
2. Student can understand use of encoders, decoders, multiplexers and demultiplexers in communication systems.
3 By learning design of combinational and sequential circuits student can understand its use in digital systems such as computers, communication systems and other modern technologies.
4 Study of ADC and DAC along with display devices will enable students to understand signal conversion and its display and their applications in digital devices.

Reference Books:

1. M. Morris Mono, "Digital logic design", Pearson Education Pvt. Ltd.
2. A Anand Kumar, "Fundamentals of digital circuits", PHI Learning Pvt Ltd.
3. A K Maini, "Digital Electronics Principles and Integrated Circuits, Wiley India Pvt Ltd.
4. R P Jain, "Modern Digital Electronics", Tata McGraw-Hill publishing company Ltd.
5. D P Kothari and J S Dhillon, "Digital Circuits and Design", Pearson Education Pvt. Ltd.

List of Experiments:

1. Study and verify the operation of AND, OR, NOT, NOR and NAND logic gates.
2. Design all basic logic gates using NOR universal gate.
3. Design all basic logic gates using NAND universal gate.
4. Verification of Demorgan's theorem.
5. Construction and verification of half adder andfull adder circuits.
6. Construction and verification of half subtractor and full subtractor circuits.
7. Design of Binary to Grey & Grey to Binary code Converters .
8. Design of BCD to excess-3 code converter.
9. Design and verification of Multiplexer circuit
10. Design and verification of De-multiplexer circuit.

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Information Technology, III-Semester

IT306 (Java Programming Lab)

Course Objectives:
1. Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
2. Understand fundamentals of object-oriented programming in Java and be familiar of the important concepts like class, inheritance and multithreading, AWT and JDBC.
3. Students will able to use the Java SDK environment to create, debug and run simple Java programs.

Unit I-Overview of Java, Installation, First Simple Program, Compilation process , Java Keywords , Identifiers , Literals, Comments, Data Types, Variables, Dynamic initialization, type conversion and casting, Operators, Control Statements.

Unit II-Declaring Objects, Introducing Methods, Constructors, this Keyword, Garbage Collection, finalize Method, Overloading Methods, Overloading Constructors, Using Objects as Parameters, Inheritance, Creating a Multilevel Hierarchy, Packages and Interfaces, Exception Handling, Multithreaded

Unit III-The Applet Class: Applet Basics, The Applet Class, Applet Architecture, Applet Initialization and Termination , Simple Applet Display Methods, Simple Banner Applet, Using the Status Window, The HTML APPLET Tag, Passing Parameters to Applets, Improving the Banner Applet.

Unit IV-Introducing the AWT: Working with Windows, Graphics, and Text, AWT Classes, Window Fundamentals, Component, Container, Panel, Frame, Working with Frame Windows, Handling Events in a Frame Window, AWT Controls, Layout Managers, and Menus, Adding and Removing Controls, Grid Layout, Border Layout, introduction to swing and servlet.

Unit V-Event Handling, Two Event Handling Mechanisms, The Delegation Event Model, Events, Event Sources, Event Listeners, Event Classes, The Mouse Event Class and others, JDBC: JDBCODBC bridge, the connectivity model, the driver manager, navigating the result set object contents, the JDBC exceptional classes, connecting to remote database.

Course Outcomes:
On the completion of this course students will be able to understand:
1. The concepts of Java programming
2. The basic terminology used in computer programming and write, compile and debug programs in JAVA language.
3. The different data types, decision structures, loops, functions to design Java programs.
4. Develop program using the java collection API as well as the java standard class library.
5. Develop Java applets

Reference Books:

1. E. Balagurusamy, "Programming with java A Primer", McGrawHill.
2. Sharanam Shah, " Core Java 8 for Beginners", Shroff Publisher.
3. Naughton & Schildt, "The Complete Reference Java 2", Tata McGraw Hill.
4. Horstmann & Cornell, "Core Java 2" (Vol I & II), Pearson.

List of Experiments:

1. Write a program that accepts two numbers from the user and print their sum.
2. Write a program to calculate addition of two number using prototyping of methods.
3. Program to demonstrate function overloading for calculation of average.
4. Program to demonstrating overloaded constructor for calculating box volume.
5. Program to show the detail of students using concept of inheritance.
6. Program to demonstrate package concept.
7. Program to demonstrate implementation of an interface which contains two methods declaration square and cube.
8. Program to demonstrate exception handling in case of division by zero error.
9. Program to demonstrate multithreading.
10. Program to demonstrate JDBC concept using create a GUI based application for student information.
11. Program to display "Hello World" in web browser using applet.
12. Program to add user controls to applets.
13. Write a program to create an application using concept of swing.
14. Program to demonstrate student registration functionality using servlets with session management.