MOSFET as an amplifier and switch

Number systems, conversion of bases, Boolean algebra, logic gates, concept of universal gate, canonical forms, Minimization using K-map

UNIT-IV

Operational Amplifier: Concept of ideal operational amplifiers, ideal op-amp parameters, inverting, non-inverting and unity gain amplifiers, adders, difference amplifiers, integrators, Other Circuits based on Operational Amplifiers 9

EXPERIMENTS

Note: Minimum Five experiments are to be performed

1. To plot the forward / Reverse Characteristics of Si P-N junction diode.
2. To plot the forward / Reverse Characteristics of Zener diode
3. Study and plot the characteristic of Zener diode as voltage regulator
4. Study of half wave rectifier and draw the nature of input / output signal. Calculate the value of Idc, Irms and ripple factor.
5. Study of Full wave rectifier and draw the nature of input / output signal. Calculate the value of Idc, Irms and ripple factor.
6. Study of Bridge Rectifier and draw the nature of input / output signal. Calculate the value of Idc, Irms and ripple factor.
7. Draw input output characteristic curve of n-p-n transistor in CE configuration
8. Draw input output characteristic curve of n-p-n transistor in CB configuration
9. Draw the drain and transfer curve of JFET
10. Study of OPAMP (741) and calculate the gain in (i)Inverting mode and (ii)Non-inverting mode
11. Study of OP-AMP as a (i) Summer (ii) Integrator (iii) Differentiator; and plot the nature of input & output waveform
12. Study of CRO and multi-meter measurement voltage, frequency, phase difference using CRO along with the testing of electronics component

Books & References

1. Electronic Devices and Circuits-Boylestad and Nashelsky, 6e, PHI, 2001
2. Electronic Devices and Circuits, A Mottershead, PHI,2000, 6e
3. Digital Computer Design, Morris Mano, PHI,2003
4. Electronic Instrumentation-H.S. Kalsi, 2e,TMH, 2007

BSM-202 Discrete Mathematics

Course category	:	Basic Sciences & Maths (BSM)
Pre-requisite Subject	:	NIL
Contact hours/week	:	Lecture: 3, Tutorial: 1, Practical: 0
Number of Credits	:	4
Course Assessment methods	:	Continuous assessment through tutorials, attendance, home assignments, quizzes and Two Minor tests and One Major Theory Examination
Course Objectives	:	The course is aimed to develop the basic mathematical skills of engineering students that are imperative for effective understanding of engineering subjects.

Course Outcomes : The students are expected to be able to demonstrate the following knowledge, skills and attitudes after completing this course

1. Use of logical notation to define different function such as set, function and relation.
2. Use of basic properties of group theory in computer science.
3. Use of induction hypotheses to prove formulae.
4. To know the basic techniques in combinatorics and counting.
5. Identify and apply properties of combinatorial structures.
6. To inculcate the habit of mathematical thinking and lifelong learning.

Topics Covered

UNIT-I 9

Set Theory, Relation and Function: Operations on sets, relations and functions, binary relations, partial ordering relations, equivalence relations, principles of mathematical induction. Finite and infinite sets, countable and uncountable sets, Cantor's diagonal argument and the power set theorem, Schröder-Bernstein theorem.

UNIT-II 9

Propositional logic: Syntax, semantics, valid, satisfiable and unsatisfiable formulas, encoding and examining the validity of some logical arguments.

Proof techniques: Forward proof, proof by contradiction, contrapositive proofs, proof of necessity and sufficiency.

UNIT-III 9

Algebraic Structures: Algebraic structures with one binary operation - semigroups, monoids and groups, congruence relation and quotient structures. Free and cyclic monoids and groups, permutation groups, substructures, normal subgroups. Algebraic structures with two binary operations - rings, integral domains and fields. Boolean algebra and Boolean ring.

UNIT-IV 9

Combinatorics: Basic counting techniques: inclusion and exclusion, pigeon-hole principle, permutation, combination, summations. Introduction to recurrence relations and generating functions.

Books & References

1. Kenneth H Rosen, Discrete Mathematics and its Applications, TMH.
2. C L Liu, Elements of Discrete Mathematics, Second Edition, Tata McGraw-Hill.
3. Bernard Kolman, Robert C Busby, and Sharon Cutler Ross, Discrete Mathematical Structures, fifth edition, Prentice-Hall of India.
4. Ralph P Grimaldi, Discrete and Combinatorial Mathematics, Pearson Education Asia.
5. J P Tremblay and R Manohar, Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw-Hill.

BCS-201 Digital Circuits and Logic Design

Course Category: PC

Pre-requisite Subject: NIL

Contact Hours/Week: Lecture: 3, Tutorial: 0 & Practical: 2

Number of Credits: 4

Course Assessment Method: Continuous Assessment through Two Tests, Teacher Assessment (Quiz, Tutorial, Assignment, Attendance), Practical Work & Viva-voce, and One Major Theory & One Practical Examinations.

Course Objective: This course helps the students in gaining the knowledge of basic principles of digital circuit design and different number systems. This course helps in building a solid foundation to undertake future courses such as computer organization and architecture.

1. Discuss basic building blocks of logic design
2. Learn how circuits are designed in a real computer system
3. Introduce different number systems
4. Learn current trends in circuit design
5. Be familiar with issues and trade-offs in the design of digital circuits

Course Outcomes The students are expected to be able to demonstrate the following knowledge, skills, and attitudes after completing this course

1. Design a finite state machine and sequential logic design.
2. Synthesize a logic design from a natural language description of a problem.
3. Realize a complete arithmetic and logic unit.
4. Generate a realization of combinational logic in a programmable gate array.
5. Simulate a complete design to evaluate functional correctness and timing.
6. Conduct an experiment to learn the logic design and prototyping process in order to acquire requisite hands-on skills

Topic Covered

UNIT-I **9L**

Binary Codes - Weighted and Non-Weighted - Binary Arithmetic Conversion Algorithms - Error Detecting and Error Correcting Codes - Canonical and Standard Boolean Expressions - Truth Tables.

UNIT-II **9L**

K-Map Reduction - Don't Care Conditions - Adders / Subtractors- Carry Look-Ahead Adder - Code Conversion Algorithms - Design of Code Converters - Equivalence Functions.

Binary/Decimal Parallel Adder/Subtractor for Signed Numbers - Magnitude Comparator - Decoders / Encoders - Multiplexers / Demultiplexers- Boolean Function Implementation using Multiplexers.

UNIT-III **9L**

Sequential Logic - Basic Latch - Flip-Flops (SR, D, JK, T and Master-Slave) - Triggering of Flip-Flops - Counters - Design Procedure - Ripple Counters - BCD and Binary - Synchronous Counters.

UNIT-IV **9L**

Registers - Shift Registers - Registers with Parallel Load - Memory Unit - Examples of RAM, ROM, PROM, EPROM - Reduction of State and Flow Tables - Race-Free State Assignment - Hazards.

Textbooks

1.Morris Mano, Digital Design, Prentice Hall of India, 2001

2.Raj Kamal, Digital Systems Principles and Design, Pearson Education, FirstEdition, 2007

3.Charles H.Roth, Jr. and Larry L. Kinney, Fundamentals of Logic Design, CL Engineering, Seventh Edition, 2013.

Reference books

1.W. H. Gothmann, Digital Electronics -An Introduction to Theory and Practice, Prentice Hall of India, 2000

2.Donald D.Givone, Digital Principles and Design, Tata McGraw –Hill, Thirteenth Impression, 2003.

BHM-202 CYBER ETHICS AND IPR (L-T-P: 2-0-0)

Course Category:	Humanities & Social Science Elective (HSSE)
Pre-requisite Subject:	None
Contact hours/week:	2 hours per week
No of Credits:	Lecture: 2, Tutorial:0, Practical: 0 (Total Credit: 02)
Course Assessment Methods:	Continuous assessment through tutorials, attendance, home assignments, quizzes and one minor test, one major theory.

Course Objective: The Course aims:
1.Sensitize course participants on ethical aspects in a broad variety of current topics of the cyber society.
2. Offer core values and virtues and methods for values-driven decisions in cyber space.
3. Empower course participants to apply values and virtues to fast developing new challenges and opportunities in cyber space.

Course Outcomes: The students are expected to be able to demonstrate the following knowledge, skills and attitudes after completing this course.

1. Recognize the main ethical aspects in cyber environment.
2. Apply ethics, core values to all decisions that are made within cyber space.
3. Analyze cyber space management approaches and models from the ethical perspective
4. To understand the cyber ethics and the laws under cyber space
5. To recognize the media laws applicable for media content and production
6. To understand the freedom of media and rights under our constitution

UNIT-I
Information Technology Act, 2000 - Aims and Objects - Overview of the Act – Jurisdiction - Electronic Governance – Electronic Evidence

Introduction to Cyber Laws: Cyber Law – National and International Perspective Cyber Law - Legal Issues and Challenges in India, Cybercrime Definition, cybercrime and information security, cybercrimes with mobile and wireless devices.

UNIT-II

Cyber Ethics, Significance of cyber-Ethics, Need for Cyber regulations and Ethics. Ethics in Information society, Introduction to Artificial Intelligence Ethics: Ethical Issues in AI and core Principles, Introduction to Block chain Ethics.

UNIT-III

Intellectual Property Rights: Definition, objectives & scope, Cyber Law and IPRs
Copyright: Definition, objectives, scope, benefits, Main Provisions of Copyright Act, Issues in Cyberspace.

UNIT-IV

Indian Patent Act, definition, objectives, types, Copyrights Vs Patent
Trademarks – introduction, benefits for business, Trademarks in Internet - Domain name registration, Trademarks in Internet

Suggested Readings

1. Stückelberger C. & Duggal P. (2018) Cyber Ethics 4.0: Serving Humanity with Values. Geneva, Globethics.net, ISBN: 978-2-88931-264-1 (online version), ISBN: 978-2-88931-265-8 (print version).
2. D.P. Mittal (Taxman Publication), Indian Patents Law and Procedure
3. B.L. Wadera, Patents, trademarks, copyright, Designs and Geographical Judications.
4. P. Narayanan (Eastern Law House), Intellectual Property Law.
5. N.S. Gopalakrishnan & T.G. Agitha, Principles of Intellectual Property (2009), Eastern Book Company, Lucknow.

BCS-202	**PRINCIPLES OF DATA STRUCTURES**
Course Category	Department Core (DC)
Pre-requisite Subject	NIL
Contact Hours/Week	Lecture: 3, Tutorial: 1 & Practical: 2
Number of Credits	5
Course Assessment Methods	Continuous Assessment through Two Tests, Teacher Assessment (Quiz, Tutorial, Assignment, Attendance), Practical Work & Viva-voce, and One Major Theory & One Practical Examinations.

Course Objective: This course helps the students in gaining the knowledge of basic principles of Data Structures. The principal objectives of this course are:

1. To provide the knowledge of basic data structures and their implementations.
2. To understand importance of data structures in context of writing efficient programs.
3. To develop skills to apply appropriate data structures in problem solving.

Course Outcomes: The students are expected to be able to demonstrate the following knowledge, skills, and attitudes after completing this course.

1. Learn the basic types for data structure, implementation, and application.
2. Know the strength and weakness of different data structures.
3. Use the appropriate data structure in context of solution of given problem.
4. Develop programming skills which require to solve given problem.
5. Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data
6. Solve problem involving graphs, trees and heaps.

Topics Covered

UNIT-I **9L**

Introduction: Basic Terminology, Elementary Data Organization, Structure Operations, Complexity and Time-Space Trade-off.

Arrays: Definition, Representation and Analysis, Single and Multi-Dimension Array, Address Calculation, Application of Arrays, Character, String in C, Character String Operation, Arrays Parameters, Ordered List, Sparse Matrices and Vectors

Stacks: Array Representation and Implementation of Stack, Operations on Stacks: Push &Pop, Array Representation of Stack, Linked Representation of Stack, Operations Associated with Stacks, Application of Stack, Conversion of Infix to Prefix and Postfix Expressions, Evaluation of Postfix Expressions using Stack, Application of Recursion in Problem like Tower of Hanoi.

UNIT-II **9L**

Queues: Array and Linked Representation and Implementation of Queues, Operations on Queue: Create, Add, Delete, Full and Empty, Circular Queues, D-Queues and Priority Queues.

Linked List: Representation and Implementation of Singly Linked Lists, Two-Way Header List, Traversing and Searching of Linked List, Overflow and Underflow, Insertion and Deletion to / from Linked Lists, Insertion and Deletion Algorithms, Doubly Linked List, Linked List in Array, Polynomial Representation and Addition, Generalized Linked List, Garbage Collection and Compaction.

UNIT-III **9L**

Trees: Basic Terminology, Binary Trees, Binary Tree Representation, Algebraic Expressions, Complete Binary Tree, Extended Binary Trees, Array and Linked Representation of Binary Trees, Traversing Binary Trees, Threaded Binary Trees, Traversing Threaded Binary Trees, Huffman Algorithm.

Binary Search Trees: Binary Search Tree (BST), Insertion and Deletion in BST, Complexity of Search Algorithm, Path Length, AVL Trees, B-Trees.

UNIT-IV **9L**

Searching and Hashing: Sequential Search, Binary Search, Comparison and Analysis, Hash Table, Hash Functions, Collision Resolution Strategies, Hash Table Implementation.

Sorting: Insertion Sort, Bubble Sorting, Quick Sort, Two Way Merge Sort, Heap Sort, Sorting on Different Keys, Practical Consideration for Internal Sorting.

Graphs: Terminology & Representations, Graphs &Multi-Graphs, Directed Graphs, Sequential Representations of Graphs, Adjacency Matrices, Traversal, Connected Component and Spanning Trees, Minimum Cost Spanning Trees.

EXPERIMENTS

Write C/C++ Programs to illustrate the concept of the following:

1. Implementation of searching and sorting techniques.
2. Implementation of list using array and linked list.
3. Implementation of push and pop operation on stack
4. Implementation of polish notation and its conversion
5. Write a program to solve the problems using iteration/recursion
6. Program for recursion removal using stack
7. Program for insertion /deletion operation on various queue & Implementation of priority queue for process scheduling
8. Program for storing data as tree structure and implementation of various traversal techniques
9. Program for storing data as graph structure and implementation of various traversal techniques
10. Program for finding shortest path in graph

TEXTBOOKS

1. Horowitz and Sahani, Fundamentals of Data Structures, Galgotia Publication, New Delhi.
2. R. Kruse et al, Data Structure and Pragram Design in C, Pearson Education Asia Delhi
3. A. M. Tenenbaum, Data Structures using C & C++, PHI, India
4. K Loudon, Mastering Algorithms with C, Shroff Publication and Distributor Pvt. Ltd.
5. Bruno R Preiss, Data Structure and Algorithms with Object Oriented Design Pattern in C++, John Wiley & Sons
6. Adam Drozdek, "Data Structures and Algorithms in C++", Thomson Asia Pvt. Ltd. Singapore

REFERENCE BOOKS

1. Lewis, H.R., Denenberg, L., Data Structures and their Algorithms. Published by AddisonWesley, UK, 1991
2. Oluwadare, S.A., Agbonifo, O.C., Fundamentals of Data structures and Algorithms. Lecture Notes, 2013

BCS-203 OBJECT ORIENTED PROGRAMMING

Course Category	PC
Pre-requisite Subject	NIL
Contact Hours/Week	Lecture: 3, Tutorial: 1 & Practical: 2

Number of Credits	5
Course Assessment Methods	Continuous Assessment through Two Tests, Teacher Assessment (Quiz, Tutorial, Assignment, Attendance), Practical Work & Viva-voce, and One Major Theory & One Practical Examinations.

Course Objective: This course helps the students in gaining the knowledge of basic principle of design and analysis of algorithms. This course helps to undertake future courses that assume as a background in data structures and algorithm design.

1. To introduce the object-oriented programming concepts.
2. To understand object-oriented programming concepts and apply them in solving problems.
3. To introduce the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes.
4. To introduce the implementation of packages and interfaces.
5. To introduce the concepts of exception handling and multithreading.
6. To introduce the design of Graphical User Interface using applets and swing controls.

Course Outcomes: The students are expected to be able to demonstrate the following knowledge, skills, and attitudes after completing this course.

1. Able to implement, compile, test and run JAVA programs comprising more than one class and to address a particular software problem.
2. Able to solve real world problems using OOP techniques.
3. To identify different components of client server architecture on Internet computing.
4. Knowledge of how to develop and deploy applications and applets in JAVA.
5. Knowledge of how to develop and deploy GUI using JAVA Swing and AWT.
6. Design, develop and implement interactive web applications.

TOPIC COVERED

UNIT-I **9L**

Introduction to the principles of object-oriented programming, Core Java: Introduction, Operator, Data types, Variables, Control Statements, Arrays, Methods & Classes, Constructors, String Handling, Inheritance, Package and Interface.

UNIT-II **9L**

Exception Handling, Multithread programming, I/O, Java Applet, Networking, Event handling, Introduction to AWT, AWT controls, Layout managers.

UNIT-III **9L**

Java Swing: Creating a Swing Applet, Labels, Text fields, Buttons, Tabbed Panes, JDBC: Connectivity Model, JDBC/ODBC Bridge, JAVA SQL package, connectivity to Remote Database, Remote method invocation (RMI).

UNIT-IV **9L**

Java Beans: Application Builder tools, The Bean Developer Kit (BDK), JAR files, Introspection, developing a simple bean, using Bound properties, The Java Beans API, Session Beans, Entity Beans, Introduction to Java Servlet: Servlet Basics, Servlet API basic, Life cycle of a Servlet, Running Servlet.

EXPERIMENTS

1. Basic programs of simple statements, conditional statements, iterative statement, and arrays.
2. Programs having object-oriented concepts like Inheritance and Interface.
3. Programs for Exception Handling and Event Handling.
4. Programs of Threads and Multithreading.
5. Programs related to Applets and Swings.
6. Program including JAVA Beans and Servlets.

TEXTBOOKS

1. Herbert Schildt, Java The Complete Reference, 9th edition, McGraw Hill Education (India) Pvt. Ltd.
2. T. Budd, Understanding Object-Oriented Programming with Java, Pearson Education
3. Balagurusamy E, "Programming in JAVA", TMH

REFERENCE BOOK

1. Deitel & Deitel, JAVA: How to Program, Pearson education
2. Margaret Levine Young, "The Complete Reference Internet", TMH.
3. Dustin R. Callway, "Inside Servlets", Addison Wesley.
4. Mark Wutica, "Java Enterprise Edition", QUE.
5. Steven Holzner, "Java2 Black book", Dreamtech.

BCS-204 IT Tools and Workshop-2

Course Category	PLBSE
Pre-requisite Subject	NIL
Contact Hours/Week	Lecture: 0, Tutorial: 0 & Practical: 4
Number of Credits	2
Course Assessment Methods	Continuous Assessment through Teacher Assessment (Quiz, Tutorial, Assignment, Attendance), Practical Work & Viva-voce, and One Practical Examinations.

Course Objective: This course helps the students in gaining the knowledge of basic tools/library being used in IT industry. The course objectives of above course are-

1. To understand the installation and various commands of Linux Operating System.
2. To understand the Shell programming and Computer Networking Concepts.
3. To understand basics of python and its important modules.

Course Outcomes: The students are expected to be able to demonstrate the following knowledge, skills, and attitudes after completing this course.

1. Understanding of Booting Process and Installation of Linux Operating System.

2. Demonstrate the usage of Shell as a programming language.
3. Understanding of Computer Networking concepts in Linux.
4. Basic knowledge of Python programming language.
5. Basic knowledge of various python libraries like xlrd, pandas, numpy, sklearn etc,
6. Understand the basic concept and structure of computer hardware and networking

EXPERIMENTS

1. Installation of Linux operating system using virtualization technique.
2. Understanding and practice of various Linux commands.
3. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
4. Illustrate by writing script that will print, message "Hello World, in Bold and Blink effect, and in different colors like red, brown etc using echo commands?
5. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
6. Illustrate by writing script using for loop to print the pyramid patterns?
7. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
8. Understanding and practice of Computer networking commands.
9. Write a python script to find factorial of a given number.
10. Write a python script to find factorial of a given number.
11. Write a Python program to count the occurrences of each word in a given string sentence.
12. Python program to create a dictionary with key as first character and value as words starting with that character.
13. Write a python program to create, append and remove lists in python.
14. Write a program to demonstrate working with tuples in python.
15. Write a python program to read excel file from xlrd module and perform processing on that file.
16. Write a program to demonstrate working with pandas module in python.
17. Write a program to demonstrate working with numpy module in python.
18. Write a program to demonstrate working with sklearn module in python.

TEXTBOOKS

1. Richard Petersen, "Linux: The Complete Reference", Sixth Edition, McGraw Hill Education.
2. E. Balaguruswamy, "Introduction to Computing and Problem Solving Using Python", McGraw Hill Education India

REFERENCE BOOK

1. Richard Blum, "Linux Command Line and Shell Scripting Bible", 4th Edition, Wiley.