|| ऋते ज्ञानान्न मुक्ती: ||

Dwarka Bahuuddeshiya Gramin Vikas Foundation's
Rajarshi Shahu College of Engineering, Buldana
Approved By AICTE New Delhi, NAAC Accredited, Affiliated to Sant Gadge Baba Amravati University

B.E. COMPUTER SCIENCE & ENGG. SEM. III & IV

Syllabus of B.E. Sem. III (Computer Science & Engineering)

3 KS01/3IT01/3KE01 ENGINEERING MATHEMATICS-III

Course Objectives:-

1. Find general solutions of linear differential equations with constant coefficients using the roots of the

2. auxiliary equation.

3. Calculate the Laplace Transform of basic functions using the definition.

4. Apply Laplace transform to find solution of linear differential equations. And solve problems related to

5. Fourier Transform

6. Compute and interpret the correlation coefficient.

7. Compute the Analytic function and Complex Analysis.

8. Perform vector differentiation and integration to analyze the vector fields and apply to compute line, surface and volume integrals.

Course Outcomes:

After successfully completing the course, the students will be able to:

1. Demonstrate the knowledge of differential equations and linear differential equations .

2. Apply Laplace transform to solve differential equations.

3. Demonstrate the use of Fourier Transform to connect the time domain and frequency domain.

4. Demonstrate the basic concepts of probability and statistics.

5. Apply the knowledge of Complex Analysis.

6. Apply the knowledge of vector calculus to solve physical problems.

SECTION-A

UNIT-I: Ordinary differential equations:- Complete solution, Operator D, Rules for finding complementary function, the inverse operator, Rules for finding the particular integral, Method of variation of parameters, Cauchy's and Legendre's linear differential equations. (7)

UNIT-II: Laplace Transform:- Definition, standard forms, properties of Laplace transform, inverse Laplace transform, Initial and final value theorem, Convolution theorem, Laplace

transform of impulse function, Unit step function, Laplace transforms of periodic function .
(7)

UNIT-III: a) Applications of Laplace Transform:- Solution of Linear differential equations, Simultaneous differential equation by Laplace transform method

b) Fourier Transform:- Definition, standard forms, Fourier transforms, properties of Fourier transforms, Convolution theorem, Fourier sine and Fourier cosine transforms and integrals, inverse Fourier transforms.(7)

SECTION-B

UNIT-IV: a) Partial differential equation of first order of following form:- (i)f $(p,q) = 0$; (ii) f $(p,q,z) = 0$;   (iii) f $(x, p) = g (y,q)$; (iv) P $p + Qq = R$ (Lagranges Form); (v) $z = px + qy + f(p,q)$ (Clairauts form)

 b) Statistics Curve fitting: Least Square Method, Coefficient of Correlations, Lines of Regression. (7)

UNIT-V: Complex Analysis: - Functions of complex variables, Analytic function, Cauchy-conditions, Harmonic function, Harmonic conjugate functions, Milne's Method, conformal mappings (translation, rotation, magnification and bilinear transformation), Expansion of function in Taylor's and Laurent's series. (7)

UNIT–VI: Vector calculus:- Scalar and vector point functions, Differentiation of vectors, Curves in space, Gradient of a scalar point function, Directional derivatives, Divergence and curl of a vector point function and their physical meaning, expansion Formulae (without proof), line, surface, volume integrals, irrotational Solenoidal Vector fields. (7)

TEXT BOOKS:

1. Elements of Applied Mathematics Vol. II by P. N. Wartikar and J.N. Wartikar,

2. Higher Engg. Mathematics by B.S. Grewal.

REFERENCE BOOKS:

1. Advancing Engg. Mathematics by E.K.Kreyzig.

2. A text book of Differential Calculus by Gorakh Prasad.

3. A Text Book of Applied Mathematics by P.N.Wartikar and J.N.Wartikar.

4. Engineering Mathematics by Ravish R Singh, Mukul Bhatt.

3KS02 DISCRETE STRUCTURE AND GRAPH THEORY

Course Pre-requisite: Basic knowledge of Mathematics

Course Objectives: Throughout the course, students will be expected to demonstrate their understanding of

Discrete Structure and Graph Theory by being able to do each of the following:

1. Use mathematically correct terminology and notation.

2. Construct correct direct and indirect proofs.

3. Use division into cases in a proof.

4. Apply logical reasoning to solve a variety of problems.

Course Outcomes : On completion of the course, the students will be able to

1. Analyze and express logic sentence in terms of predicates, quantifiers, and logical connectives.

2. Derive the solution for a given problem using deductive logic and prove the solution based on logical inference.

3. Classify algebraic structure for a given mathematical problem.

4. Perform combinatorial analysis to solve counting problems.

5. Develop the given problem as graph net works and solve with techniques of graph theory

Unit I: Foundations: Logic and Proofs (Hours: 7)

Propositions, Truth Tables, Compound Propositions, Logical Operators, Logic and Bit Operations; Logical Equivalences, De Morgan's Laws, Predicates, Quantifiers: Restricted Domains, Precedence, Logical Equivalences; Rules of Inference for Propositional Logic, Use to Build Arguments, Resolution, Combination for Propositions and Quantified Statements; Proofs Terminology, Methods, Direct Proofs, Proof by Contraposition and Contradiction;

Unit II: Sets, Functions and Relations (Hours: 7)

Introduction, Venn Diagrams, Subsets, Size of a Set, Power Sets, Cartesian Products, Set Notation with Quantifiers, Truth Sets and Quantifiers, Set Operations; Inverse Functions, Compositions and Graphs of Functions, Important Functions, Partial Functions; Sequences, Recurrence Relations, Special Integer Sequences, Summations; Countable Sets, An Uncountable Set; Functions as Relations, Relations on a Set, Properties of Relations, Combining Relations; n-ary Relations, Operations on n-ary Relations; Representing Relations Using Matrices; Closures, Transitive Closures

Unit III: Number Theory and Induction (Hours: 6)

Division, The Division Algorithm, Modular Arithmetic, Arithmetic Modulo m; Primes, Trial Division, Conjectures and Open Problems About Primes, GCD and LCM, The Euclidean Algorithm, gcds as Linear Combinations; Linear Congruences, The Chinese Remainder Theorem, Fermat's Little Theorem, Pseudoprimes, Primitive Roots and Discrete Logarithms, Applications: Hashing Functions, Mathematical Induction and Examples of Proofs, Mistaken Proofs, Guidelines for Proofs; Strong Induction, Examples of Proofs.

Unit IV: Algebraic Structures (Hours: 7)

Algebraic Systems: Examples and General Properties; Semigroups and Monoids: Homomorphism of Semigroups and Monoids, Subsemigroups and Submonoids; Groups: Definitions, Subgroups and Homomorphisms, Cosets and Lagrange's Theorem, Normal Subgroups, algebraic Systems with Two Binary Operations.

Unit V: Counting (Hours: 7)

Basic Counting Principles, Complex Counting Problems, Subtraction and Division Rule, The Pigeonhole Principle, The Generalized Pigeonhole Principle, Applications; Permutations, Combinations, Generating Permutations, Generating Combinations.

Unit VI: Graphs (Hours: 6)

Graph Models; Basic Terminology, Special Simple Graphs, Bipartite Graphs, Matchings, Applications of Special Types of Graphs, New Graphs from Old; Graph Representation, Adjacency and Incidence Matrices, Isomorphism of Graphs, Determining Isomorphism; Paths, Connectedness in Undirected Graphs and Directed Graphs, Paths and Isomorphism, Counting Paths Between Vertices; Euler Paths and Circuits, Hamilton Paths and Circuits, Applications of Hamilton Circuits; Planar Graphs: Euler's Formula, Kuratowski's Theorem; Graph Coloring: Introduction, Applications of Graph Colorings.

Text Book: Kenneth H. Rosen: Discrete Mathematics and Its Applications, 7thEdition, McGraw-Hill.

Reference Books:

1. J. P. Tremblay and R. Manohar: Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw-Hill Edition, McGraw-Hill.

2. Norman L. Biggs: Discrete Mathematics, 2nd Edition, Oxford University Press.

3. Seymour Lipschutz and Marc Lars Lipson: Schaum's Outline of Theory and Problems of Discrete Mathematics, 3rd Edition, Schaum's Outlines Series, McGraw-Hill.

4. C. L. Liu and D. P. Mohapatra: Elements of Discrete Mathematics: A Computer Oriented Approach, 3rd Edition, Tata McGraw-Hill, McGraw-Hill.

3KS03 OBJECT ORIENTED PROGRAMMING

Course Pre-requisite: Computer Programming

Course Objectives:

1. To explore the principles of Object Oriented Programming (OOP) such as data abstraction, encapsulation, inheritance and polymorphism.

2. To use the object-oriented paradigm in program design.

3. To Provide programming insight using OOP constructs.

4. To lay a foundation for advanced programming

Course Outcomes : On completion of the course, the students will be able to

1. Apply Object Oriented approach to design software.

2. Implement programs using classes and objects.

3. Specify the forms of inheritance and use them in programs.

4. Analyze polymorphic behaviour of objects.

5. Design and develop GUI programs.

6. Develop Applets for web applications

Unit I: Introduction to Object Oriented Programming (Hours:7)

Introduction, Need of OOP, Principles of Object-Oriented Languages, Procedural Language Vs OOP, Application of OOP, Java Virtual Machine, Java features, Program Structures. Java Programming Constructs: Variables, Primitive data types, Identifier, Literals, Operators, Expressions, Precedence Rules and Associativity, Primitive Type Conversion and Casting, ,Flow of Control.

Unit II: Classes and Objects (Hours:7)

Classes, Objects, Creating Objects, Methods, Constructors, Cleaning up Unused Objects, Class Variable and Methods, this keyword, Arrays, Command Line Arguments.

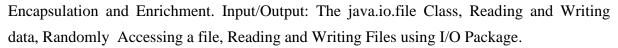Unit III:Inheritance, Interfaces and Packages (Hours:6)

Inheritance: Inheritance vs. Aggregation, Method Overriding, super keyword, final keyword, Abstract class. Interfaces: Defining interfaces, Implementing interfaces, Accessing interface variables, Extending interfaces. Packages: Packages, java.lang package, Enum type.

Unit IV: Exception handling and Input /Output (Hours:7)

Exception: Introduction, Exception handling Techniques, User-defined exception, Exception

Encapsulation and Enrichment. Input/Output: The java.io.file Class, Reading and Writing data, Randomly Accessing a file, Reading and Writing Files using I/O Package.

Unit V: Applets (Hours:7)

Introduction, Applet Class, Applet structure, Applet Life cycle, Common Methods used in displaying the output, paint (), update () and repaint (), More about applet tag, get Document Base() and get Code Base () methods, Applet Context Interface, Audio clip, Graphic Class, Color, Font, Font Metrics.

Unit VI: Unit Title: Event Handling (Hours:6)

Introduction, Event delegation Model, java.awt.event Description, Sources of events, Event Listeners, Adapter classes, Inner Classes. Abstract Window Toolkit: Introduction, Components and Containers, Button, Label, Checkbox, Radio Buttons, List Boxes, Choice Boxes, Textfield and Textarea, Container Class, Layouts, Menu, Scrollbar.

Text Books:

1. Sachin Malhotra and Saurabh Choudhary: Programming in Java, Oxford University Press 2010.
2. Herbert Schildt: Java Complete References (McGraw Hill)

Reference Books:

1. H.M.Dietel and P.J.Dietel, "Java How to Program" Pearson Education/PHI, Sixth Edition.
2. E. Balagurusamy: Programming with Java (McGraw Hill)
3. Dr. R. NageswaraRao: Core Java An Integrated Approach (Dreamtech)
4. Khalid Mughal: A Programmer's Guide to Java Certification, 3rdEdition (Pearson)
5. Sharnam Shah and Vaishali Shah: Core Java for Beginners, (SPD), 2010.

3KS04/3KE04 DATA STRUCTURES

Course Pre-requisite: Fundamentals of programming Language & Logic Building Skills

Course Objectives:

1. To understand the linear and nonlinear data Structures and its memory representations.

2. To perform different operations on data structures such as insertion, deletion, searching and traversing.

3. To understand various data searching and sorting methods with its complexity.

4. To introduce various techniques for representation of the data in the real world.

Course Outcomes: On completion of the course, the students will be able to

1. Apply various linear and nonlinear data structures

2. Demonstrate operations like insertion, deletion, searching and traversing on various data structures

3. Examine the usage of various structures in approaching the problem solution.

4. Choose appropriate data structure for specified problem domain

Unit I: Introduction to Data Structures (Hours: 7)

Introduction to Data structures, Data Structure Operations, Algorithmic Notation, Complexity of algorithms.String processing: storing strings, character data type, string operations, word processing, and pattern matching algorithms.

Unit II: Array & Record Structure (Hours: 7)

Linear arrays : Memory Representation of arrays, traversing linear arrays, insertion & deletion operations, Bubble sort, Linear search and Binary search algorithms. Multi dimensional arrays, Pointer arrays.Record structures and Matrices.

Unit III: Linked lists (Hours: 6)

Linked lists: Memory Representation of Linked List, traversing a linked list, searching a linked list. Memory allocation & garbage collection.Insertion & deletion operations on linked lists. Header linked lists, Two- way linked lists.
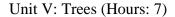
Unit IV: Stack & Queue (Hours: 7)

Stacks: Sequential Memory Representation of Stack, Arithmetic expressions: Polish notation. Quick sort, Recursion, Tower of Hanoi.

Queues: Sequential Memory Representation of Queue, DeQueue, Priority queues.

Unit V: Trees (Hours: 7)

Introduction to Trees, Binary trees, Memory Representation of Binary Tree, Traversing binary trees, Header nodes, Binary Search Tree, Heap and heap sort, Path length & Huffman's algorithm.

Unit VI: Graphs & Sorting Algorithms (Hours: 6)

Introduction to Graphs, Memory representation of graphs, Warshalls' algorithm, operations on Graphs, Breadth First Search, Depth First Search. Sorting : Insertion Sort, Selection Sort, Radix sort, Merge Sort.

Text Books:

1. Seymour Lipschutz: Data Structures, Schaum's Outline Series, McGraw-Hill, International Editions.

2. Trembley, Sorenson: An Introduction to Data Structures with Applications, McGraw Hill.

Reference Books:

1. Ellis Horowitz, Sartaj Sahni: Fundamentals of Data Structures, CBS Publications.

2. Data Structure Using C, Balagurusamy.

3. Standish: Data Structures in Java, Pearson Education.

|| ऋते ज्ञानान्न मुक्ती: ||

Dwarka Bahuuddeshiya Gramin Vikas Foundation's
Rajarshi Shahu College of Engineering, Buldana
Approved By AICTE New Delhi, NAAC Accredited, Affillated to Sant Gadge Baba Amravati University

3KS05 ANALOG& DIGITAL ELECTRONICS

Course Prerequisite:Basic Physics.

Course Objectives:

1. To get the introductory knowledge of PN Junction Diode, Bipolar Junction Transistor, Field Effect Transistor.

2. To understand number systems and conversion between different number systems.

3. To get basics knowledge about digital ICs and digital systems.

4. To study the design of combinational circuits and sequential circuits

Course Outcomes : At the end of course students will able to

1. Explain basic concepts of semiconductor devices and its application.

2. Compare different Number System and basics of conversion of number systems.

3. Realize different minimization technique to obtain minimized expression.

4. Design Combinational Circuits.

5. Design and Develop Sequential Circuits.

Unit I: PN Junction Diode and Bipolar Junction Transistor (Hours: 7)

PN-Junction Diode, Characteristics and Parameters, BJT operation, BJT Voltages and Currents, BJT Amplification: Current and Voltage, BJT Switching, Common-Base Characteristics, Common-Emitter Characteristics, Common- Collector Characteristics

Unit II: Field Effect Transistors (Hours: 7)

Junction Field Effect Transistors, n-Channel and p-Channel JFET, JFET Characteristics, JFET Parameters, FET Amplifications and Switching, MOSFETs: Enhancement MOSFET, Depletion_Enhancement MOSFET, Comparison of p-channel and n-channel FETs, Introduction to CMOS.

Unit III: Number System (Hours: 6)

Binary Number System, Signed and unsigned Number, Octal Number System, Hexadecimal Number System, Conversions between Number Systems, r's and (r-1)'s Complements Representation, Subtraction using 1's and 2's Complements, BCD, Gray Code, Excess 3 Code and Alpha numeric codes.

Unit IV: Minimization Techniques (Hours: 7)

Logic Gates, Boolean Algebra, Logic Operation, Axioms and Laws of Boolean Algebra, Reducing Boolean Expression, Boolean Functions and their representation, SOP Form, POS Form, Karnaugh Map (up to 5 variable), Limitation of Karnaugh Map, Quine- McCluskey Minimization Technique (up to 5 variable).

Unit V: Combinational Circuits (Hours: 7)

Introduction, Design Procedure, Adders, Subtractors, Binary Parallel Adder, 4 Bit Parallel Subtractor,   Look-ahead-carry Adder, BCD adder, BCD Subtractor, Multiplexer, De-multiplexer, Decoder,   Encoder, Comparator, Parity bit Generator/Checkers, Boolean Expression Implementation using these ICs.

Unit VI: Sequential Circuits (Hours: 6)

Flip-flops: S-R, J-K, Master slave J-K, D-type, T-type, Flip flop Excitation Table, Conversion of Flip Flops, Registers: SISO, SIPO, PISO, PIPO, Universal Shift Register.

Counters: Asynchronous and Synchronous counter, Up/Down counter, MOD-N counter, Ring counter, Johnson counter.

Text Books:

1. David A. Bell: "Electronic Devices and Circuits", 5e , Oxford University Press.
2. Jain R.P. "Modern Digital Electronics", 3e, TMH.

Reference Books:

1. Millman & Halkies: "Electronic Devices & Circuits", 2e, McGraw Hill.
2. Sedra& Smith: "Microelectronics Circuits", 5e, Oxford University Press.
3. Anand Kumar: "Switching Theory and Logic Design", 3e, PHI Learning Private Limited
4. Wakerly, "Digital Design: Principles and Practices", 3 e, Pearson Education, 2004.

3KS06 OBJECT ORIENTED PROGRAMMING - LAB

Course Pre-requisite: Basic Computer Programming

Course Objectives: Design, implement, test, and debug simple programs in an object-oriented programming language.

1. To develop the knowledge of object-oriented paradigm in the Java programming language.

2. To evaluate classical problems using java programming.

3. To develop software development skills using java programming for real world applications.

Course Outcomes : On completion of the course, the students will be able to

1. Design, implement, test, and debug simple programs in an object-oriented programming language.

2. Interpret the basics of object-oriented design and the concepts of encapsulation, abstraction, inheritance, and polymorphism

3. Build applications in Java by applying concepts like interfaces, packages and exception handling.
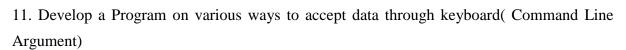
4. Make use of Java concepts like API, Applets, AWT.

List of Experiments:

This is a sample list of Experiments; minimum 12 experiments are to be performed covering the entire syllabus. At least two experiments should be beyond syllabi based on learning of syllabi (Apply)

1. Introduction to Object Oriented Programming and installation of JDK. Write a program to print a message "Hello World..."

2. Develop a program to explain use of Operators in java.

3. Develop a Program to study and implement Looping Statements belonging to Java.

4. Develop a Program to study and implement Selection Statements belonging to Java.

5. Develop a program to study and implement some Pyramid.

6. Develop a program to demonstrate the concept of Class, Method and Object.

7. Develop a program to study and implement the concept of Method Overloading.

8. Develop a program to study and implement concept of Constructor in Java.

9. Develop a program to study and implement concept of Constructor Overloading in Java.

10. Develop a program to study and implement the Array in Java.

11. Develop a Program on various ways to accept data through keyboard(Command Line Argument)

12. Develop a program to study and implement the concept of Inheritance.

13. Develop a program to study and implement the concept of Method Overriding.

14. Develop a program to study and implement the Abstract Class.

15. Develop a program to study and implement the concept of Interface in Java.

16. Develop a program to study and implement Exception Handling Mechanism in Java.

17. Develop a program to study and implement Java I/O.

18. Develop a program to study and implement simple Applet in java.

19. Develop a program on Applet to demonstrate Graphics, Font and Color class.

20. Develop a Program on passing parameters to applets

21. Develop a Program to create GUI application without event handling using AWT controls

22. Develop a Program to create GUI application with event handling using AWT controls

23. Develop a program on Multithreading

24. Develop a Program to create GUI application with event handling using Swing controls

25. Mini Project based on content of the syllabus. (Group of 2-3 students)

3KS07 DATA STRUCTURE - LAB

Course Pre-requisite: Basics of programming Language & Logic Building Skills

Course Objectives:

1. To understand the linear and nonlinear data Structures and its memory representations.

2. To perform different operations on data structures such as insertion, deletion, searching and traversing.

3. To understand various data searching and sorting methods with its complexity.

4. To introduce various techniques for representation of the data in the real world.

Course Outcomes : On completion of the course, the students will be able to

1. Apply various linear and nonlinear data structure.

2. Demonstrate operations like insertion, deletion, searching and traversing on various data structures

3. Examine the usage of various structures in approaching the problem solution.

4. Choose appropriate data structure for specified problem domain

List of Experiments:

This is a sample list of Experiments; minimum 12 experiments are to be performed covering the entire syllabus. At least two experiments should be beyond syllabi based on learning of syllabi (Apply)

1. Write a program to find out largest number from the array and also find it's location.

2. Write a program to traverse an array and find the sum and average of data elements from an array.

3. Write a Program to a) insert an element in an array b)delete an element from an array.

4. To study and execute the Linear search method

5. To study and execute the Binary Search method

6. To study and execute the Pattern matching Algorithms(Slow and Fast)

7. To study and execute Bubble sort method.

8. To study and implement various operations on singly linked list

(a) Traversing the linked list.

(b) Insert a node at the front of the linked list.

(c) Delete a last node of the linked list.

(d) Searching a Linked list.

9. To study and implement following operations on the doubly linked list.

(a) Insert a node at the front of the linked list.

(b) Insert a node at the end of the linked list.

(c) Delete a last node of the linked list.

(d) Delete a node before specified position.

10. To study and implement following operations on the circular linked list.

(a) Insert a node at the end of the linked list.

(b) Insert a node before specified position.

(c) Delete a first node of the linked list.

(d) Delete a node after specified position.

11. Understand the stack structure and execute the push, pop operation on it.

12. Understand the Queue structure and execute the insertion, deletion operation on it.

13. Formulate and demonstrate Transforming Infix Expressions to Postfix Expression using Stack.

14. Formulate and demonstrate the Evaluation of Postfix Expression using Stack.

15. To study and execute Quick sort method.

16. Understand the Tree structure and implement the Pre-order, In-order, post-order traversingoperations on it.

17. Understand the concept of Recursion and write a program to calculate factorial of a numberusing Recursion.

18. Understand the Heap sort and implement it on given data.

19. Understand the Insertion sort and implement it on given data.

20. Understand the Selection sort and implement it on given data.

21. To study and execute Merge sort method.

22. To study and execute Radix sort method.

23. Write a Program to implement the concept of BFS algorithm.

24. Write a Program to implement the concept of DFS algorithm.

25. To study and execute Josephus problem.

3KS08 ANALOG & DIGITAL ELECTRONICS - LAB

Course Pre-requisite: Students should have the knowledge of Basic Physics.

Course Objectives:

1. To impart the concepts of analog and digital electronics practically.

2. To provide students basic experimental experiences in the operation of semiconductor device and Digital ICs.

3. To learn the operation of various logic gates and their implementation using digital IC's.

4. To learn the realization of various combinational and sequential circuits.

Course Outcomes : After successfully completing the lab, the students will be able to

1. Apply practically the concepts of analog and digital electronics.

2. Explain the operation and characteristics of semiconductor devices.

3. Illustrate the operation of various logic gates and their implementation using digital IC's.

4. Design and implement various combinational logic circuits.

5. Design and implement various sequential logic circuits

List of Experiments:

This is a sample list of Experiments; minimum 10 experiments are to be performed covering the entire syllabus. At least two experiments should be beyond syllabi based on learning of syllabi (Apply)

1. To study V-I characteristics of a PN Junction diode in Forward and Reverse bias.

2. To Sketch and Study the input and output characteristics of transistor connected in Common Emitter (CE) configuration..

3. To Sketch and Study the input and output characteristics of transistor connected in Common Base (CB) configuration

4. To Sketch and Study the input and output characteristics of transistor connected in Common Collector (CC) configuration.

5. To plot static characteristics of FET & calculate its parameters gm, rd and μ.

6. To implement Logic gates using TTL ICs (7400, 7402, 7404, 7408, 7410, 7411, 7420, 7427, 7432, 7486).

7. Study and verify the truth table of half adder and full adder using logic gates.

8. Study and verify the truth table of half subtractor and full subtractor using logic gates

9. To compare two 4 bits number and verify the output using 4-bit comparator IC 7485.

10. Implementation of 4×1 multiplexer using logic gates.

11. Implementation and verification of Demultiplexer and Encoder using logic gates.

12. Implementation of 4bit parallel adder using 7483 IC.

13. Design and verify the 4 bit synchronous counter.

14. Design and verify the 4 bit asynchronous counter.

15. Verification of truth table of SR, JK, T and D Flip Flops.

List of Experiments beyond syllabus:

1. Design and Implementation of Op-amp as an inverting amplifier.

2. Design and Implementation of Op-amp as a non-inverting amplifier.

3. To design and find frequency of A stable multi-vibrator using IC 555.

3KS09 C SKILL - LAB - I

Course Prerequisite: Basic knowledge of any Programming Language

Course Objectives:

1. To be able to program design with functions using Python.

2. To understand data and information processing techniques.

3. To understand to Design a program to solve the problems.

4. To be able to access database using python programming.

5. To be able to design web applications using python programming.

Course Outcomes : On completion of the course, the students will be able to

1. Describe the Numbers, Math functions, Strings, List, Tuples and Dictionaries in Python

2. Interpret different Decision Making statements, Functions, Object oriented programming in Python

3. Summarize different File handling operations

4. Explain how to design GUI Applications in Python and evaluate different database operations

5. Develop applications using Django framework or Flask

List of Experiments:

This is a sample list of Experiments, minimum 12 experiments are to be performed covering the entire syllabus. At least two experiments should be beyond syllabi based on learning of syllabi (Apply)

1. Write python program to store data in list and then try to print them.

2. Write python program to print list of numbers using range and for loop

3. Write python program to store strings in list and then print them.

4. Write python program in which an function is defined and calling that function prints Hello World.

5. Write a python script to print the current date in the following format "Sun May 29 02:26:23 IST 2017"

6. Write a program to create, append, and remove lists in python.

7. Write a program to create, concatenate and print a string and accessing sub-string from a given string.

8. Write a program to demonstrate working with tuples in python.

9. Write a program to demonstrate working with dictionaries in python.

10. Write a python program to find largest of three numbers.

11. Write python program in which an function(with single string parameter) is defined and calling that function prints the string parameters given to function.

12. Write python program in which an class is define, then create object of that class and call simple print function define in class.

13. Write a Python script that prints prime numbers less than 20.

14. Write a python program to find factorial of a number using Recursion.

15. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.

16. Write a script named copyfile.py. This script should prompt the user for the names of two text files.
The contents of the first file should be input and written to the second file.

17. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.

18. Write a Python class to convert an integer to a roman numeral.

19. Write a Python class to implement pow(x, n)

20. Write a Python class to reverse a string word by word.

21. Accessing and working with databases using Python.

22. Create data frame from .csv files and operations on it.

23. Plotting various graphs using Python.

24. Developing basic GUI using Python.

25. Developing web applications using Django framework or Flask

Reference Books :

1. "Core Python Programming", R. NageswaraRao, dreamtech press.

2. "Python Programming A Modular Approach With Graphics, Database, Mobile and WebApplications", SheetalTaneja, Naveen Kumar, Pearson.

3. Python Web Development with Django By Jeff Forcier, Paul Bissex, Wesley J Chun, Addison-Wesley Professional.

4. Kenneth A. Lambert, The Fundamentals of Python: First Programs, 2011, Cengage Learning

5. Allen B. Downey , " Think Python: How to Think Like a Computer Scientist", Second Edition, Shroff/O'Reilly Publishers

6. John V Guttag. "Introduction to Computation and Programming Using Python", Prentice Hall of India

7. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, "Data Structures and Algorithms in

Python", Wiley

8. Introduction to Computation and Programming using Python, by John Guttag, PHI Publisher, Revised and

Expanded version (Referred by MIT)

****

|| ॠते ज्ञानान्न मुक्ती: ||

Dwarka Bahuuddeshiya Gramin Vikas Foundation's
Rajarshi Shahu College of Engineering, Buldana
Approved By AICTE New Delhi, NAAC Accredited, Affiliated to Sant Gadge Baba Amravati University

SEMESTER - IV

4KS01 ARTIFICIAL INTELLIGENCE

Course Pre-requisite: Basic concepts of Data Structures, Algorithms, Programming

Course Objectives:

1. To present an overview of Artificial Intelligence (AI) principles and approaches.

2. To understand the historical evolution of Artificial Intelligence.

3. To learn various searching techniques and identify to address a particular problem).

Course Outcomes : On completion of the course, the students will be able to

1. Explain concepts of Artificial Intelligence and different types of intelligent agents and their architecture.

2. Formulate problems as state space search problem & efficiently solve them.

3. Summarize the various searching techniques, constraint satisfaction problem and example problems - game playing techniques.

4. Apply AI techniques in applications which involve perception, reasoning and learning.

5. Compare the importance of knowledge, types of knowledge, issues related to knowledge acquisition and representation.

Unit I: Introduction to AI (Hours: 7)

Introduction : What Is AI?, The Foundations of Artificial Intelligence, The History of Artificial

Intelligence, The State of the Art, Risks and Benefits of AI, Intelligent Agents: Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The Structure of Agents

Unit II: Problem Solving Through AI (Hours: 7)

Introduction, Representation the AI Problems, Production System, Algorithm of Problem Solving, Examples of AI Problems, Nature of AI Problems

Unit III: Uninformed Search Strategies (Hours: 6)

Problem-Solving Agents, Example Problems, Search Algorithms, Uninformed Search Strategies: Breadth-First Search, Uniform-Cost Search, Depth First Search, Bidirectional Search, Depth Limited Search, Iterative Deepening Depth-First Search

Unit IV: Informed Search Strategies (Hours: 7)

Basic Concept of Heuristic Search and Knowledge, Designing of Heuristic Function, Heuristic Search Strategies: Generate-And-Test, Best-First Search, Problem Reduction, Hill Climbing, Constraint Satisfaction, Means-Ends-Analysis

Unit V: Adversarial Search & Games (Hours: 7)

Game Theory, Optimal Decisions in Games, Mini-Max Search, Alpha Beta Pruning, Additional Refinements, Monte Carlo Tree Search, Stochastic Games, Partially Observable Games, Limitations of Game Search Algorithms

Unit VI: Introduction to Knowledge (Hours: 6)

Introduction, Types of Knowledge, Knowledge Representation, Knowledge Storage, Knowledge Acquisition, Knowledge Organization and Management, Basic Concepts of Knowledge Engineering

Text Books:

1. Artificial Intelligence: A Modern Approach by Stuart Russell & Peter Norvig (Pearson - 4 th Ed.)

2. Artificial Intelligence by Ela Kumar (IK International Publishing House Pvt. Ltd.)

Reference Books:

1. Artificial Intelligence by Elaine Rich and Kevin Knight (Tata McGraw Hill - 3rd Ed.)

2. A First Course in Artificial Intelligence by Deepak Khemani (Tata McGraw Hill - 1st Ed.)

3. Artificial Intelligence and Expert Systems by Patterson (PHI)

4. Introduction to Artificial Intelligence by RajendraAkerkar (PHI Learning Pvt. Ltd.)

4KS02 DATA COMMUNICATION AND NETWORKING

Course Prerequisite: Computer and Data Communication Requirements

Course Objectives:

1. To understand the building blocks of digital communication system.

2. To prepare mathematical background for communication signal analysis.

3. To understand and analyze the signal flow in a digital communication system

4. To analyze error performance of a digital communication system in presence of noise and other interferences.

5. To evaluate the errors using various error detection & correction techniques.

6. To understand network based protocols in data communication and networking.

Course Outcomes : On completion of the course, the students will be able to

1. Describe data communication Components, Networks, Protocols and various topology based network architecture

2. Design and Test different encoding and modulating techniques to change digital –to-digital conversion, analog-to-digital conversion, digital to analog conversion, analog to analog conversion,

3. Explain the various multiplexing methods and evaluate the different error detection & correction techniques.

4. Illustrate and realize the data link control and data link protocols.

5. Describe and demonstrate the various Local area networks and the IEEE standards.

Unit I: Introduction to Data Communication (Hours: 7)

Introduction: Data Communication, Components, Networks, Network types: Local Area Network,Wide Area Network, Switching, The Internet, Accessing the Internet, Standards and Administration: Internet Standards, Internet Administration, Network Models:TCP/IP Protocol Suite,TheOSI Model, Transmission media: Introduction, Guided media & Unguided media-Wireless. Switching: Introduction, Circuit Switched Networks, Packet Switching.

Unit II: Data link Layer (Hours: 6)

Data Link Layer: Introduction, Nodes & Links, Services , Two categories of link, Two sub-layers, Error detection and correction: Introduction, Block Coding, Cyclic codes, Checksum, Forward Error Correction, Data link control: DLC services, Data-Link Layer Protocol, HDLC, Point-To-Point Protocol, Media Access Control (MAC): Random Access, Controlled Access, Channelization.

|| ऋते ज्ञानान्न मुक्ती: ||

Dwarka Bahuuddeshiya Gramin Vikas Foundation's
Rajarshi Shahu College of Engineering, Buldana
Approved By AICTE New Delhi, NAAC Accredited, Affiliated to Sant Gadge Baba Amravati University

Unit III: Network Layer (Hours: 7)

Introduction to Network layer Network Layer Services: Packetizing, Routing and Forwarding, Other Services Packet Switching: Datagram Approach: Connectionless Service, Virtual-Circuit Approach: Connection-Oriented Service , Network Layer performance: Delay, Throughput, Packet Loss, Congestion Control, IPV4 Address: Address Space, Classful Addressing, Classless Addressing, Dynamic Host Configuration Protocol (DHCP), Network Address Resolution (NAT), Forwarding of IP packets: Forwarding Based on Destination Address, Forwarding Based on Label, Routers as Packet Switches

Unit IV: Network Layer Protocol (Hours: 7)

Network Layer Protocols: Internet Protocol (IP),Datagram Format, Fragmentation, Security of IPv4 Datagrams,ICMPV4: Messages, Debugging Tools, ICMP Checksum,Mobile IP: Addressing, Agents, Three Phases, Inefficiency in Mobile IP,Routing algorithms: Distance Vector routing, Link State Routing, IPV6 Addressing: Representation, Address Space, Address Space Allocation, Auto configuration, Renumbering, Transition from IPV4 to IPV6: Strategies, Use of IP Addresses

Unit V: Transport Layer (Hours: 6)

Introduction to Transport layer: Introduction, Transport-Layer Services, Connectionless and ConnectionOriented Protocols, Transport-Layer Protocols: Simple Protocol, Stop-and-Wait Protocol, Go-Back-N Protocol (GBN), Selective-Repeat Protocol, Bidirectional Protocols: Piggy backing, User Datagram Protocols: User Datagram, UDP Services, UDP Applications, Transmission Control Protocol: TCP Services, TCP Features , Segment, A TCP Connection, State Transition Diagram, Windows in TCP, Flow Control, Error Control, TCP Congestion Control, TCP Timers, Options, SCTP: SCTP Services, SCTP Features

Unit VI: Application layer (Hours: 7)

Introduction to Application layer: Providing Services, Application-Layer Paradigms, Client-Server Programming: Application Programming Interface, Using Services of the Transport Layer, Iterative Communication Using UDP, Iterative Communication Using TCP, Concurrent Communication, World wide web and HTTP: World Wide Web, Hyper-Text Transfer Protocol (HTTP) FTP: Two Connections, Control Connection, Data Connection, Security for FTP, Electronic Mail: Architecture, Web-Based Mail, E-Mail Security, Domain Name System (DNS):Name Space, DNS in the Internet, Resolution, Caching, Resource Records, DNS Messages, Registrars, Security of DNS, Network Management: Introduction. Configuration Management, Fault Management, Performance Management, Security Management, Accounting Management, SNMP: Managers and Agents, Management Components, ASN.1: Language Basics, Data Types, Encoding.

Text Book: Behrouz A. Forouzan: Data Communication and Networking, (5/e) (TMH).

Reference Books:

1. William Stallings: Data & Computer Communications, 6/e, Pearson Education

2. William L. Schweber : Data Communication, McGraw Hill

3. J.Freey : Computer Communication & Networks, AEW Press

4. D. Corner: Computer Networks & Internet, Pearson Education.

4KS03 OPERATING SYSTEM

Course Pre-requisite: Discrete Structures, Data Structure, Any programming Language

Course Objectives:

1. To make students aware of the kernel and shell structure of the operating systems.

2. To make students aware of the purpose, structure and functions of operating systems

3. To equip students with understanding of the various scheduling algorithms in OS.

4. To make students aware of understanding of memory management in different OS.

Course Outcomes : On completion of the course, the students will be able to

1. Explain memory management issues like external fragmentation, internal fragmentation.

2. Illustrate multithreading and its significance.

3. List various protection and security mechanisms of OS.

4. Analyze and solve the scheduling algorithms.

5. Analyze the deadlock situation and resolve it.

6. Compare various types of operating systems

Unit I: Introduction to OS (Hours: 7)

Introduction: Operating System definition, OS Evolution, Components and Services, Process Concept, Process Scheduling, Operations on Processes, Cooperating Processes, Interprocess Communication, Threads Overview, Multithreading Models, Threading Issues, Java Threads

Unit II: Process Scheduling (Hours: 7)

Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR, Priority, Multilevel Queue, Multilevel Feedback Queue Scheduling

Unit III: Process Synchronization (Hours: 6)

Process Synchronization Basics: The Critical-Section Problem, Synchronization Hardware, Semaphores, Monitors, Deadlocks: Definition & Characterization, Deadlocks Prevention, Avoidance, Detection and Recovery from Deadlock

Unit IV: Memory Management (Hours: 7)

Memory Management Background, Swapping, Contiguous Memory Allocation Schemes, Paging, Segmentation, Virtual Memory Management: Background, Demand paging scheme, Process Creation, Page Replacement Policies, Allocation of Frames, Thrashing

Unit V: Unit Title: File System (Hours: 7)

File-System Interface; Directory Structure, File-System Mounting, File Sharing &Protection, File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management. File Recovery

Unit VI: Unit Title: I/O System (Hours:6)

I/O Systems : Overview, I/O Hardware, Application I/O Interface, Kernel I/O Subsystem, Transforming I/O to Hardware Operations , Disk Scheduling, Disk Management, Swap-Space Management, RAID Structure.

Text Book : Avi Silberschatz, P.B.Galvin, G.Gagne: "Operating System Concepts" (9/e) John-Wiley & Sons.

Reference Books:

1. A.S.Tanenbaum "Modern Operating Systems" Pearson Education.

2. William Stallings "Operating Systems" Prentice-Hall.

3. D. M. Dhamdhere "Operating Systems" Tata McGraw-Hill.

4. P. Balkrishna Prasad: "Operating Systems" Scitech Publications (I) Pvt. Ltd.

|| ऋते ज्ञानान्न मुक्ती: ||

Dwarka Bahuuddeshiya Gramin Vikas Foundation's
Rajarshi Shahu College of Engineering, Buldana
Approved By AICTE New Delhi, NAAC Accredited, Affiliated to Sant Gadge Baba Amravati University

4KS04 MICROPROCESSOR & ASSEMBLY LANGUAGE PROGRAMMING

Course Pre-requisite: Computer Programming and Computer Fundamentals

Course Objectives:

1. To explore 8086 microprocessor and its architecture.

2. To introduce interfacing techniques of 8086 microprocessor.

3. To introduce basics of Internet of ThingsCourse Outcomes : On completion of the course, the students will be able to

1. Describe 8086 microprocessor and its architecture; also understand instruction processing during the fetch-decode-execute cycle.

2. Design and Test assembly language programs using 8086 microprocessor instruction set.

3. Demonstrate the implementation of standard programming constructs, including control structures and functions, in assembly language.

4. Illustrate and realize the Interfacing of memory & various I/O devices with 8086 microprocessor.

5. Explain the basic concepts of Internet of Things

Unit I: 8086 Architecture (Hours: 7)

8086 architecture and pin configuration, Software model of 8086 microprocessor. Memory addresses space and data organization. Data types. Segment registers, memory segmentation. IP & Data registers, Pointer, Index registers. Memory addresses generation.

Unit II: 8086 Instruction Set (Hours: 7)

8086 Instruction set overview, addressing modes. 8086 instruction formats. 8086 programming: Integerinstructions and computations: Data transfer instructions, Arithmetic instructions and their use in 8086 programming.

Unit III: 8086 Instruction Set (Hours: 6)

8086 programming: logical instructions. Shift and rotate instructions and their use in 8086 programming. 8086 flag register and Flag control instructions, compare instruction, control flow and jump instructions, Loops & loop handling instructions. 8086 programming using these instructions.

Unit IV: Subroutines& Macros (Hours: 7)

The 8086 stack segment and stack related instructions. 8086 I/O Address space.Subroutines and related instructions, Parameter passing, Concept of Macros, Status saving on stack.Concept of recursion at assembly program level.8086 Programming using subroutines, recursion and macros.

Unit V: 8086 Interrupt (Hours: 7)

8086 Interrupts types, priority and instructions. Interrupt vector table, External hardware-interrupt interface signals & interrupts sequence. Software interrupts. Non-maskable interrupts. 8086 microprocessor interrupt programming.

Unit VI: Internet of Things (IoT) (Hours: 6)

Internet of things: An overview, IoT conceptual framework, IoT Architectural View, Technology behind IoT, Sources of IoT, M2M communication, Examples of IoT.

Text Book:

1. A. K. Ray & K. M. Bhurchandi: Advanced Microprocessors & Peripherals, Third Edition (TMH).

2. Raj Kamal: Internet of Things, Architecture and Design Principals, McGraw Hill Education (India) Private Limited

Reference Books:

1. W. A. Triebel& Avatar Singh: The 8088/8086 Microprocessors (4e) (PHI /Pearson Education)

2. Liu & Gibson: The 8088/8086 Microprocessor Architecture Programming and Interface (6/e) (PHI)

|| ऋते ज्ञानान्न मुक्ती: ||

Dwarka Bahuuddeshiya Gramin Vikas Foundation's
Rajarshi Shahu College of Engineering, Buldana
Approved By AICTE New Delhi, NAAC Accredited, Affiliated to Sant Gadge Baba Amravati University

4KS05 THEORY OF COMPUTATION

Course Pre-requisite: Discrete Mathematics, Data Structures

Course Objectives:

1. To understand different automata theory and its operation.

2. To understand mathematical expressions for the formal languages

3. To study computing machines and comparing different types of computational models

4. To understand the fundamentals of problem decidability and Un-Decidability

Course Outcomes: On completion of the course, the students will be able to

1. To construct finite state machines to solve problems in computing.

2. To write regular expressions for the formal languages.

3. To construct and apply well defined rules for parsing techniques in compiler

4. To construct and analyze Push Down, Turing Machine for formal languages

5. To express the understanding of the Chomsky Hierarchy.

6. To express the understanding of the decidability and un-decidability problems.

Unit I: Finite State Machines (Hours: 8)

Alphabet, String, Formal and Natural Language, Operations, Definition and Design DFA (Deterministic Finite Automata), NFA (Non Deterministic Finite Automata), Equivalence of NFA and DFA: Conversion of NFA into DFA, Conversion of NFA with epsilon moves to NFA, Minimization Of DFA, Definition and Construction of Moore and Mealy Machines, Inter-conversion between Moore and Mealy Machines.Minimization of Finite Automata. (Construction of Minimum Automaton)

Unit II: Regular Expression and Regular Grammar (Hours: 8)

Definition and Identities of Regular Expressions, Construction of Regular Expression of the given Language, Construction of Language from the RE, Conversion of FA to RE using Arden's Theorem, Inter-conversion RE to FA, Pumping Lemma for RL, Closure properties of RLs(proofs not required), Regular grammar, Equivalence of RG (RLG and LLG) and FA.

Unit III: Context Free Grammar and Languages (Hours: 8)

Introduction, Formal Definition of Grammar, Notations, Derivation Process: Leftmost Derivation, Rightmost Derivation, Derivation Trees, Construction of Context-Free Grammars and Languages, Pumping Lemma for CFL, Simplification of CFG, Normal Forms (CNF and GNF), Chomsky Hierarchy.

Unit IV: Pushdown Automata (Hours: 8)

Introduction and Definition of PDA, Construction of PDA, Acceptance of CFL, Equivalence of CFL and PDA: Inter-conversion , Introduction of DCFL and DPDA, Enumeration of properties of CFL, Context Sensitive Language, Linear Bounded Automata.

Unit V: Turing Machines (Hours: 8)

Formal definition of a Turing Machine, Design of TM, Computable Functions, Church's hypothesis, Counter machine, Variants of Turing Machines: Multi-tape Turing machines, Universal Turing Machine.

Unit VI: Decidability and Un-Decidability (Hours: 8)

Decidability of Problems, Halting Problem of TM, Un-Decidability: Recursive enumerable language, Properties of recursive & non-recursive enumerable languages, Post Correspondence Problem, Introduction to Recursive Function Theory

Text Books:

1. Hopcraft H.E. & Ullman J: Introduction to Automata Theory, Languages and Computation

2. Peter Linz: An Introduction to Formal Languages and Automata

Reference Books:

1. Rajesh K. Shukla: Theory of Computation, CENGAGE Learning, 2009.

2. K V N Sunitha and N Kalyani: Formal Languages and Automata Theory, McGraw Hill,2010

3. Lewis H.P. and Papadimition C.H.: Elements of Theory of Computation

4. Mishra &Chandrashekharan: Theory of Computation

5. C.K.Nagpal: Formal Languages and Automata Theory, Oxford University Press, 2011.

6. VivekKulkarni : Theory of Computation, OUP India, 2013.

4KS06 DATA COMMUNICATION & NETWORKING LAB

Course Pre-requisite: Computer and Data Communication Requirements

Course Objectives:

1. To understand the working principle of various communication protocols

2. To understand and analyze the signal flow in a digital communication system.

3. To analyze error performance of a digital communication system in presence of noise and other interferences.

4. To evaluate the errors using various error detection & correction techniques.

5. To understand network based protocols in data communication and networking.

Course Outcomes : On completion of the course, the students will be able to

1. Analyze performance of various communication protocols

2. Implement Configure various network protocols.

3. Compare IP Address classes of networks

List of Experiments:

This is a sample list of Experiments; minimum 12 experiments are to be performed covering the entire syllabus. At least two experiments should be beyond syllabi based on learning of syllabi (Apply)

1. To study various LAN topologies and their creation using network devices, cables and computers. .

2. To connect the computers in Local Area Network.

3. Familiarization with Networking Components and devices: LAN Adapters, Hubs, Switches, Routers etc.

4. Write a program of bit stuffing used by Data Link Layer

5. Write a program to implement CRC(Cyclic Redundancy Check)

6. Write a program to implement Checksum

7. Write a program to implement Sliding window

8. Configure Internet connection and use IP-Config, PING / Tracer and Net stat utilities to debug the network issues.

9. Configuration of TCP/IP Protocols in Windows and Linux.

10. Transfer files between systems in LAN using FTP Configuration, install Print server in a LAN and share the printer in a network.

11. Write a C Program to determine if the IP Address is in Class A, B, C, D, or E

12. Write a C Program to translate Dotted Decimal IP Address into 32 Bit Address.

13. Configure Host IP, Subnet Mask and Default Gateway in a System in LAN(TCP/IP Configuration)

|| ऋते ज्ञानान्न मुक्तीः ||

Dwarka Bahuuddeshiya Gramin Vikas Foundation's
Rajarshi Shahu College of Engineering, Buldana
Approved By AICTE New Delhi, NAAC Accredited, Affiliated to Sant Gadge Baba Amravati University

4KS07 OPERATING SYSTEM - LAB

Course Pre-requisite: Basic computer programming

Course Objectives:

1. To make students aware of the kernel and shell structure of the operating systems.

2. To make students aware of the purpose, structure and functions of operating systems

3. To equip students with understanding of the various scheduling algorithms in OS.

4. To make students aware of understanding of memory management in different OS.

Course Outcomes : On completion of the course, the students will be able to

1. Explain memory management issues like external fragmentation, internal fragmentation.

2. Illustrate multithreading and its significance.

3. List various protection and security mechanisms of OS.

4. Analyze and solve the scheduling algorithms.

5. Analyze the deadlock situation and resolve it.

6. Compare various types of operating systems

List of Experiments:

This is a sample list of Experiments, minimum 12 experiments are to be performed covering the entire syllabus. At least two experiments should be beyond syllabi based on learning of syllabi (Apply)

1. To study Linux Operating System along with its installation.

2. To Study and Execute basic file commands and process related open source Ubuntu commands

a. Commands to view all executing, block and suspended process.

b. Command to check and change the priority of process CPU utilization for executing processes.

c. Commands to check for child process, sub-processes, process tree, abort & end process and all other basics commands related to processes

3. Write a program for multithreading using C.

4. To simulate First Come First Serve & Shortest Job First process scheduling algorithm

5. To simulate Shortest Job First process scheduling algorithm

6. To simulate Preemptive Shortest Job First process scheduling algorithm

7. To implement Round Robin Process scheduling Algorithm

8. To implement Priority Based Process scheduling Algorithm

9. To implement and analyze multi-level queue scheduling algorithm

10. To implement the following file allocation strategies.

11. To simulate paging technique of memory management.

12. To implement theFIFO page replacement policy

13. To implement the LRU page replacement policy

14. To implement the optimalpage replacement policy

15. To simulate producer-consumer problem using semaphores.

16. To implement Dining-Philosophers problem to deal with concurrency control mechanism.

17. To implement contiguous memory allocation strategies to detect fragmentation using: First Fit, Best Fit and Worst Fit.

18. To implement FCFS Disk Scheduling algorithm

19. To implement SCAN Disk Scheduling algorithm

20. To implement C-SCAN Disk Scheduling algorithm

21. To simulate Bankers algorithm for deadlock avoidance

22. To implement following memory management techniques

Implement MVT and MFT where memory block size is 100 for 5 processes. Enter no. of blocks for each process and calculate internal fragmentation.

23. To simulate LFU page replacement algorithms

24. To simulate the Single level directory file organization techniques.

25. To Simulate bankers algorithm for Dead Lock Avoidance (Banker's Algorithm)

Dwarka Bahuuddeshiya Gramin Vikas Foundation's
Rajarshi Shahu College of Engineering, Buldana
Approved By AICTE New Delhi, NAAC Accredited, Affiliated to Sant Gadge Baba Amravati University

|| ऋते ज्ञानान्न मुक्तीः ||

4KS08 MICROPROCESSOR & ASSEMBLY LANG. PROG. - LAB

Course Pre-requisite: Computer Programming, Number System

Course Objectives: In this lab student will learn about' Microprocessor and Interfacing' in regards to digital computer, microprocessor architecture, programming with 8086 microprocessor and different peripherals.

Course Outcomes On completion of the course, the students will be able to

1. Analyze the internal workings of the microprocessor

2. Design and develop programs in Assembly Language Programming

3. Describe 8086 microprocessor and its architecture; also understand instruction processing during the fetch-decode-execute cycle.

4. Design and Test assembly language programs using 8086 microprocessor instruction set.

5. Demonstrate the implementation of standard programming constructs, including control structures and functions, in assembly language

6. Illustrate and realize the Interfacing of memory & various I/O devices with 8086 microprocessor.

List of Experiments:

This is a sample list of Experiments; minimum 12 experiments are to be performed covering the entire syllabus. At least two experiments should be beyond syllabi based on learning of syllabi (Apply)

1. Installation and Introduction of TASM Assembler.

2. Write a program for addition of two 8-bits numbers and two 16-bits numbers.

3. Write a program for subtraction of two 8-bits numbers and two 16-bits numbers.

4. Write a program for multiplication of two 8-bits numbers.

5. Write a program for division of two 8-bits numbers

6. Write a program to check whether a given number is even or odd.

7. Write a program to demonstrate Logical Group and Shift Rotate Instructions.

8. Write a program to check whether a given number is positive or negative.

9. Write a program to find greatest of two 8-bits signed &unsigned numbers.

10. Block Transfer Program

11. Write a program to find Factorial of a number using loop instruction.

12. Write a program to find cube of a given number using Subroutine.

13. Write a program to find square of a given number using Subroutine.

14. Write a program to find square of a given number using Macro.

15. Write a program to find whether the string is palindrome or not.

16. To convert BCD Number Program

17. Write a program to perform Reverse of the String

18. Write a program to transfer 10-bytes from one memory bank to another memory bank.

19. Program for sorting an array for 8086 microprocessor.

20. To write an assembly language program to arrange the given numbers in descending order.

21. Program for searching for a number/character in a string for 8086 microprocessor.

4KS09 C-SKILL-LAB II

Course Pre-requisite: Basic knowledge of scripting language, Programming language.Basic understanding of Electronic concepts.

Course Objectives: To develop an ability to design and implement static and dynamic website and to develop

embedded systems with the help of Raspberry Pi/Ardino.

Course Outcomes : On completion of the course, a student will be able to

1. Develop client server program and web applications

2. Make use of project-based experience for web application development.

3. Create embedded systems using Raspberry Pi/Ardino

List of Experiments:

This is a sample list of experiments, minimum 12 experiments are to be performed covering the entire syllabus. At least two experiments should be beyond syllabi based on learning of syllabi (Apply)

1. Introduction to PHP and configure it to work with Apache Web Server.

2. Design web pages for your college containing a description of the courses, departments, faculties, library etc, use href, list tags.

3. Create your class timetable using table tag.

4. Create user Student feedback form (use textbox, text area , checkbox, radio button, select box etc.)

5. Create your resume using HTML tags also experiment with colors, text , link , size and also other tags you studied.

6. Design a web page of your home town with an attractive background color, text color, an Image, font etc. (use internal CSS).

7. Develop a JavaScript to display today's date.

8. Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.

9. Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).

10. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.

11. Write a PHP program to display a digital clock which displays the current time of the server.

12. Write the PHP programs to do the following: a. Implement simple calculator operations. b. Find the transpose of a matrix.

13. Write a PHP program to sort the student records which are stored in the database using selection sort.

14. Study and Install IDE of Arduino and different types of Arduino.

15. Write program using Arduino IDE for Blink LED.

16. Write Program for RGB LED using Arduino.

17. Study the Temperature sensor and write a Program for monitor temperature using Arduino.

18. Study and Implement RFID, NFC using Arduino. □ Study and implement MQTT protocol using Arduino.

19. Study and Configure Raspberry Pi.

20. WAP for LED blink using Raspberry Pi.

21. Study and Implement Zigbee Protocol using Arduino / Raspberry Pi.

22. Create Smart Plugs with Arduino and Raspberry Pi.

23. Interfacing digital sensors with raspberry pi.

24. Creating a webpage to control I-O devices, Reading data from sensor and passing to web page.

25. Implement a program to access Analog sensor via wifi with HTML Web server.