ESTD: 1980

SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE (AUTONOMOUS)
(Affiliated to Andhra University, Visakhapatnam), (Recognised by AICTE, New Delhi)
Accredited by NAAC with 'A' Grade
Recognised as Scientific and Industrial Research Organisation
CHINNA AMIRAM (P.O):: BHIMAVARAM :: W.G.Dt., A.P., INDIA :: PIN: 534 204

SCHEME OF INSTRUCTION & EXAMINATION
(Regulation R16)

II/IV B.TECH
(With effect from **2016-2017** Admitted Batch onwards)
Under Choice Based Credit System

COMPUTER SCIENCE AND ENGINEERING

I-SEMESTER

Code No.	Course	Credits	Lecture Hrs	Tutorial Hrs	Lab Hrs	Total Contact Hrs/Week	Sessional Marks	Exam Marks	Total Marks
B16 CS 2101	Data Structures	4	3	1	--	4	30	70	100
B16 EC 2103	Elements of Electronics Engineering	4	3	1	--	4	30	70	100
B16 ENG 2102	Discrete Mathematical Structures	4	3	1	--	4	30	70	100
B16 CS 2102	Object Oriented Programming	4	3	1	--	4	30	70	100
B16 CS 2103	Digital Logic Design	4	3	1	--	4	30	70	100
B16 ENG 2103	Environmental Studies	2	3	1	--	4	30	70	100
B16 CS 2105	Data Structures Lab.	2	--	--	3	3	50	50	100
B16 CS 2106	Object Oriented Programming Lab.	2	--	--	3	3	50	50	100
B16 ENG 2104	English Proficiency	2	1	1	--	2	50	50	100
B16 ENG 2105	Industry Oriented Training.	1	--	--	2	2	50	--	50
Total		**29**	**19**	**7**	**8**	**34**	**380**	**570**	**950**

II/IV B.Tech-I Sem
CSE

Code: B16 CS 2101

DATA STRUCTURES

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. Be familiar with basic techniques of algorithm analysis
2. Master the implementation of linked data structures such as linked lists and binary trees
3. Be familiar with advanced data structures such as balanced search trees, priority queues.
4. Be familiar with several sub-quadratic sorting algorithms including quick sort, merge sort and heap sort
5. Master analyzing problems and writing program solutions to problems using the above techniques.

Course Outcomes:

1. Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms .
2. Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs . Write programs that use arrays, records, linked structures, stacks, queues, trees, and graphs Demonstrate different methods for traversing trees [ABET (a)].
3. Compare alternative implementations of data structures with respect to performance [ABET (a, b, c)].
4. Compare and contrast the benefits of dynamic and static data structures implementations [ABET (a, b, c)].
5. Describe the concept of recursion, give examples of its use, describe how it can be implemented using a stack [ABET (a, c)].
6. Discuss the computational efficiency of the principal algorithms for sorting, searching.

SYLLABUS

Basic Concepts
System Life Cycle, Algorithm Specification, Recursive Algorithms, Data Abstraction, Performance Analysis, Space Complexity, Time Complexity, Asymptotic Notation, Comparing Time Complexities.

Arrays and Structures
Array as an Abstract Data Type, Polynomial Abstract Data Type, Introduction to Sparse Matrix, Sparse Matrix Abstract Data Type, Transposing a Sparse Matrix, Sparse Matrix Multiplication, Representation of Multidimensional Arrays, Structures and Unions, Internal Implementation of Structures, Self-Referential Structures.

Recursion, Simple Searching and Sorting Techniques
Recursive functions, Introduction to Searching, Sequential Search, Binary Search, Interpolation Search, Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Introduction to Merge Sort, Iterative Merge Sort, Recursive Merge Sort, Shell Sort.

Stacks and Queues
Stack Abstract Data Type, Queue Abstract Data Type, Stacks and Queues using arrays, Mazing Problem, Introduction to Evaluation of Expressions, Evaluating Postfix Expressions, Infix to Postfix and Prefix conversion, Multiple Stacks and Queues, Circular Queues using arrays.

Linked Lists
Pointers, Dynamically Allocated Storage using pointers, Singly Linked Lists, Dynamically Linked Stacks and Queues, Polynomials, Representing Polynomials as Singly Linked Lists, Adding Polynomials, Erasing Polynomials, Polynomials as Circularly Linked Lists, Additional List Operations, Operations for Singly Linked Lists, Operations for Doubly Linked Lists, Radix Sort.

Trees
Representation of Trees, Binary Trees Abstract Data Type, Properties of Binary Trees, Binary Tree Representations, Binary Tree Traversals, Additional Binary Tree Operations, Threaded Binary Trees, Heap Abstract Data Type, Priority Queues, Insertion into a max heap, Deletion from a max heap, Heap Sort, Introduction to Binary Search Trees, Searching a Binary Search Tree, Inserting an Element into a Binary Search Tree, Deleting an Element from a Binary Search Tree, Height of a Binary Search Tree, Counting Binary Trees.

Graphs
Graph Abstract Data Type, Definitions, Graph Representations, Elementary Graph Operations, Depth First Search, Breadth First Search, Connected Components, Spanning Trees, Minimum Cost Spanning Trees, Prim's and Kruskal's Algorithms, Shortest Paths and Transitive Closure, Single Source All Destination - Dijkstra's Algorithm, All Pairs Shortest Paths - Floyd's Algorithm, Transitive Closure using Warshall's Algorithm.

Text Books:

1. Fundamentals of Data Structures in C, 2nd edition, Horowitz, Sahani and Anderson-Freed, Universities Press, 2008.

Reference Books:

1. Data Structures using C by Aaron M. Tenenbaum,Y. Langsam and M.J. Augenstein, Pearson Education, 2009.
2. Data Structures with C by Seymour lipschutz, Schaum Outline series, 2010.
3. Data Structures using C by R. KrishnaMoorthy G. Indirani Kumaravel, TMH, New Delhi,2008.

Code: B16 EC 2103

ELEMENTS OF ELECTRONICS ENGINEERING
(Common to CSE & IT)

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. To give the exposure to the students on semiconductor physics of the intrinsic and extrinsic semiconductors.
2. To give the exposure to the students on the basics of semiconductor diodes, special purpose diodes like Zener diode, Photo diode, LED and tunnel diode.
3. To give the exposure to the students on rectifies circuits using diodes.
4. To give the exposure to the students on basics of BJT, JFET and MOSFET and biasing of BJT and FETs
5. To give the exposure to the students on the analysis of transistor at low and high frequencies.

Course Outcomes:

After completion of the course the students will be able to

1. Understand the physical structure, principles of operation, electrical characteristics and circuit models of diodes, BJTs and FETs
2. Use this knowledge to analyze and design basic electronic application circuits.
3. Extend the understanding of how electronic circuits and their functions fit into larger electronic systems.

SYLLABUS

Transport phenomenon in semiconductors :
Intrinsic and Extrinsic semiconductors, Charge densities in semiconductors, Drift and Diffusion currents, Half effect, Mass action law.

P-N Junction Diode :
Basic operation and V-I characteristics of semiconductor diode, diode current equation, Zener diode, LED, Photo diode and tunnel diode *(Introductory treatment only)*

Diode Rectifiers :
Half wave and full wave rectifiers with and without filters, Bridge Rectifier expressions – Ripple factor, Efficiency, capacitor filters

Bipolar Junction Transistor :
Introduction, construction, basic operation, modes of operation-Active , cutoff and saturation, Transistor circuit configurations- CE, CB, CC – input and output Characteristics in various configurations *(Introductory treatment only*

Transistor Biasing and Thermal Stabilization :
Transistor Biasing, Thermal runaway, Stabilization, Different methods of Biasing- Fixed bias, Collector feedback bias, self bias, Bias Compensation.

Transistor Amplifiers :
CE, CB, CC amplifier h-parameter model for Transistor amplifier

Field Effect Transistors :
Junction Field effect Transistors (JFET)- JFET characteristics, JFET parameters, JFET biasing, MOSFETS- Depletion and Enhancement MOSFET.

Text Books :

1. Electronic Device and Circuits by Sanjeev Guptha

Reference Books :

1. Integrated Electronics – Millman & Halkias

Code: B16 ENG 2102

DISCRETE MATHEMATICAL STRUCTURES
(Common to CSE & IT)

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. To understand mathematical arguments using logical connectives and quantifiers and verify the validity arguments using propositional, predicate logic & truth tables.
2. To know about the concepts of counting techniques
3. To know how to solve recurrence relations.
4. To understand various types of relations and discuss varies properties of the relations.
5. To understand the concepts in graphs and trees.
6. To learn about Groups, Lattices and their properties, Boolean algebra and its importance in computer field and minimization of Boolean expressions.

Course Outcomes:

At the end of the course students will be able to

1. Rewrite the mathematical arguments using logical connectives and quantifiers and verify the validity of the arguments using propositional and predicate logic.
2. Solve different counting problems.
3. Solve the recurrence relations which occur in many fields.
4. Identify and give examples of various types of relations and describe various properties of relations.
5. Determine isomorphism of graphs and utilize the concepts in graphs & trees in their fields.
6. Understand the importance of Groups, lattice structures and their diagrammatic representations and also the importance of Boolean algebra in computer science.

SYLLABUS

Introduction :
Sets – Operations on sets – Logic : Logical inferences, Methods of proof of an implications– First order logic and other proof methods-Rules of inference for quantified propositions-mathematical induction.

Elementary Combinatorics & Recurrence relations :
Basics of counting – Combinations and Permutations – their enumeration with and without repetition - Principle of Inclusion and Exclusion and its applications, Generating functions of sequences - Calculating their coefficients-Recurrence relations-solving recurrence relations-method of characteristic roots-Non-homogeneous recurrence relations and their solutions.

Relations and Diagraphs :

Relations and directed graphs-Special properties of binary relations-equivalence relations-Ordering relations-operations on relations-Paths and closures-Directed graphs and Adjacency matrices.

Graphs Theory :

Basic concepts – Isomorphism – sub graphs - planar graphs - Euler's formula -Multi graphs and Euler Circuits - Hamiltonian graphs – Graph coloring and Chromatic number – Four color theorem - Trees and their properties – definitions of different tree structures.

Groups :

Definitions of Binary operation, Algebraic Structure, Semi-group, Monoid, Group and Abelian group.

Lattices :

Lattices and Properties of lattices – lattices as partially ordered sets – sublattices – Direct product and Homomorphisms - Isomorphisms – Modular lattices Distributive lattices – Complemented lattices.

Boolean Algebra :

Definition – Sub algebra – Direct product – Homomorphisms – Isomorphisms – Boolean functions – Representation of Boolean functions – Minimizations of Boolean functions using K-maps.

Text Books :

1. For the first four topics, scope and treatment as in "Discrete Mathematics for computer scientists and mathematicians" by Joe. L .Mott, Abraham Kandel & T.P. Baker, Prentice Hal of India Ltd, New Delhi.

2. For the topics five and six, scope and treatment as in "Discrete mathematical structures with applications to computer science" by J.P. Trembly & R. Manohar, Tata McGraw-Hill Publishing company, New Delhi.

Reference Books :

1. "Discrete mathematics and its applications" by Keneth. H. Rosen, Tata McGraw-Hill Publishing Company, New Delhi.
2. "Discrete Mathematics" by Richard Johnson Baug, Pearson Education, New Delhi.
3. "Discrete and Combinatorial Mathematics" by Ralph. G. Grimaldi, Pearson Education, New Delhi.

Code: B16 CS 2102

OBJECT ORIENTED PROGRAMMING

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

On completing this course student will be able to

1. Understand the syntax and principles of Object oriented programming language, and to programs using control statements, classes and interfaces.
2. Design and development of secure and extendable C++ applications.
3. Understanding the concepts of oops, different predefined classes and packages Understand the concepts of polymorphism

Course Outcomes:

1. Students will be able to handle I/O streams and Run time errors.
2. Students will be able to construct applications and Identify where data structures are appearing in them

SYLLABUS

Part 1: C++

Basics of Object Oriented Programming:
Object Oriented Paradigm, Principles of OOP, benefits of OOP, data types, declarations, expressions and operator precedence, functions, scope of variables.

Introduction to C++:
Classes and objects, Constructors & Destructors, constructor with dynamic allocation, explicit constructor, Operator Overloading through Unary, Binary, Assignment and Stream operators & type conversions.

Inheritance and Manipulating Strings:
Derived classes, syntax of derived classes, making private members inheritable, single, multilevel, multiple, hierarchical, hybrid inheritance, Virtual base Class, abstract classes, Creating String Objects, Manipulating String Objects, Relational Operations, Accessing String Characteristics.

Polymorphism:

Pointers, virtual functions and polymorphism- pointers to objects, this pointer, pointers to derived classes, virtual and pure virtual functions, Dynamic polymorphism, Virtual destructor, Virtual Base Class, Dynamic Casting, Cross Casting, Down Casting.

Templates, Exception handling, Streams and Files in C++:

Class templates, Function templates, member function templates, exception handling, managing console I/O operations, Stream Classes, Formatted and Unformatted i/o operations, managing output with manipulators, working with files.

Part 2: JAVA

Introduction to JAVA:

Introduction, Classes and Objects, Inheritance, Arrays, strings and Vectors, Exception Handling, Managing I/O files in Java.

Packages and Interface, and Multi threading: Packages, Interfaces, creating, threads, thread states, thread methods, exceptions, priority in threads, synchronization, Runnable interface, life cycle of an Applet.

Text Books:

1. Object oriented Programming using C++: E. Balagurusamy, PHI.
2. Programming with JAVA- A primer: E. Balagurusamy, PHI.

Reference Books:

1. JAVA 2.0- Complete Reference: Herbert Schildt & F. Naughton.
2. Introduction to JAVA PROGRAMMING by Y.Daniel Liang (PHI)
3. Object Oriented Programming in C++: N. Barkakati, PHI
4. Object Oriented Programming through C++ by Robat Laphore.
5. Object Oriented Analysis and Design by Andrew Haigh – (Tata Mcgrah Hjill.)

Code: B16 CS 2103

DIGITAL LOGIC DESIGN

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. To introduce the basic principles for design of combinational circuit and sequential circuits.
2. To learn simple digital circuits in preparation for computer engineering.

Course Outcomes:

A student who successfully fulfills the course requirements will have demonstrated:

1. An ability to define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.
2. An ability to understand the different Boolean algebra theorems and apply them for logic functions.
3. An ability to define the Karnaugh map for a few variables and perform an algorithmic reduction of logic functions.
4. An ability to define the following combinational circuits: multiplexer, de-multiplexers encoders/decoders, comparators, arithmetic-logic units; and to be able to build simple circuits.
5. An ability to understand asynchronous and synchronous sequential circuits, like counters and shift registers.
6. An ability to understand memories like RAM and ROM, Programmable Logic Array and Programmable Array Logic.

SYLLABUS

Binary Systems:
Digital Systems. Binary Numbers. Number Base Conversions. Octal and Hexadecimal Numbers. Complements. Signed Binary Numbers. Binary Codes. Binary Storage and Registers. Binary Logic.

Boolean Algebra and Logic Gates:
Basic Definitions. Axiomatic Definition of Boolean Algebra. Basic Theorems and Properties of Boolean Algebra. Boolean Functions. Canonical and Standard Forms. Other Logic Operations. Digital Logic Gates. Integrated Circuits.

Combinational Logic Design, Gate-Level Minimization:
The Map Method. Four- Variable Map. Five-Variable Map. Product of Sums Simplification. Don't-Care Conditions. NAND and NOR Implementation. Other Two- Level Implementations. Exclusive-OR Function. Hardware Description Language(HDL).

Combinational Logic:
Combinational Circuits. Analysis Procedure. Design Procedure. Binary Adder-Subtractor.DecimalAdder.BinaryMultiplier.MagnitudeComparator.Decoders. Encoders. Multiplexers. HDL For Combinational Circuits.

Sequential Logic Design, Synchronous Sequential Logic:
Sequential Circuits .Latches .Flip-Flops. Analysis of Clocked Sequential Circuits. HDLForSequentialCircuits.StateReductionAssignment.DesignProcedure.

Registers and Counters:
Registers. Shift Registers. Ripple Counters. Synchronous Counters. Other Counters. HDL for Registers and Counters.

Memory and Programmable Logic:
Introduction. Random-Access Memory. Memory Decoding, Error Detection and Correction. Read-Only Memory. Programmable Logic Array. Programmable Array Logic. Sequential Programmable Devices.

Text Books:

1. Digital Design, 3rdEdition, M.Morris Mano, Pearson Education.

Reference Books:

1. Digital Logic Design Principles, Norman Balabanian & Bradley Carlson, John Wiley & Sons(Asia) Pvt.Ltd.,2002.
2. Fundamentals of Digital Logic with VH.

Code: B16 ENG 2103

ENVIRONMENTAL STUDIES
(Common to CIV, CSE & IT)

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period.**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs**	**Credits**	**: 2**

Course Objectives:

Students learn

1. To develop an awareness and sensitivity to the total environment and its related problems.
2. To participate actively participation in environmental protection and improvement.
3. To develop skills for active identification and development of solutions to environmental problems
4. To evaluate environment programmes in terms of social, economic, ecological and aesthetic factors.
5. To Create a "CONCERN AND RESPECT FOR THE ENVIRONMENT"

Course Outcomes:

Students will be able to

1. Get awareness among the students about the nature and natural ecosystems.
2. Learn sustainable utilization of natural resources like water, land, minerals, air.
3. Learn resource pollution and over exploitation of land, water, air and catastrophic (events) impacts of climate change, global warming, ozone layer depletion, marine, radioactive pollution etc to inculcate the students about environmental awareness and safe transfer of our mother earth and its natural resources to the next generation.
4. Safe guard against industrial accidents particularly nuclear accidents.
5. Learn Constitutional provisions for the protection of natural resources.

SYLLABUS

Global Environmental Crisis:

Environmental Studies - Definition, Scope and importance, Need for public awareness. Global Environmental Crisis

Ecosystems:

Basic concepts, Forest Ecosystems, Grassland Ecosystems and Desert Ecosystems, Aquatic Ecosystems

Biodiversity:

Introduction to Biodiversity**,** Value of Bio-diversity, Bio-geographical classification of India, India as a Mega-diversity habitat, Threats to biodiversity, Conservation of Biodiversity: In-situ and Ex-situ conservation of bio-diversity.

Environmental and Natural Resources Management:

Land Resources: Land degradation, soil erosion and desertification, Effects of modern agriculture **Forest Resources:** Use and over exploitation-Mining and Dams-their effects on forest and tribal people, **Water resources**: Use and over utilization of surface and ground water, Floods, droughts, conflict over water, water logging and salinity, dams – benefits and problems

Energy Resources: Renewable and non-renewable energy sources, use of alternate energy sources-impact of energy use on environment.

Environmental Pollution:

Causes, Effects and Control measures of - Air pollution, Water pollution, Soil pollution, Marine Pollution, Thermal pollution, Noise pollution, Nuclear Hazards; Climate change and global warming, acid rain and Ozone layer depletion.

Environmental Problems in India:

Drinking water, Sanitation and Public health, population growth and environment; Water Scarcity and Ground Water Depletion; Rain water harvesting, Cloud seeding and Watershed management.

Text Books:

1. Environmental Studies (From Crisis to Cure) by R. Rajagopalan, Oxford university Press, 2008
2. Environmental Studies by Anubha Kaushik & C.P. Kauskik, New Age International (P) Ltd, New Delhi, 2006

Reference Books:

1. Environmental Sciences by G.Tyler Miller, JR,10^{th} ed, Thomson publishers, 2004

Code: B16 CS 2105

DATA STRUCTURES LAB

	Sessionals : 50
Lab : 3 Periods	**Ext. Marks : 50**
Exam : 3 Hrs.	**Credits : 2**

Course Objectives:

1. To implement stacks and queues using arrays and linked lists.
2. To develop programs for searching and sorting algorithms.
3. To write programs using concepts of various trees.
4. To implement programs using graphs.

Course Outcomes:

1. Student will be able to write programs to implement stacks and queues.
2. Ability to implement various searching and sorting techniques.
3. Ability to implement programs using trees and graphs.

SYLLABUS

Implement the following programs using C-Language.

1. Write a program for sorting a list using Bubble sort and then apply binary search.
2. Write a program to implement the operations on stacks.
3. Write a program to implement the operations on circular queues.
4. Write a program for evaluating a given postfix expression using stack.
5. Write a program for converting a given infix expression to postfix form using stack.
6. Write a program for implementing the mazing problem.
7. Write a program for the representation of polynomials using linked list and for the addition of two such polynomials
8. Write a program for quick sort
9. Write a program for Merge sort.
10. Write a program for Heap sort
11. Write a program to create a binary search tree and for implementing the in order, preorder, post order traversal using recursion
12. Write a program for finding the transitive closure of a digraph
13. Write a program for finding the shortest path from a given source to any vertex in a digraph using Dijkstra‘s algorithm
14. a)Write a program for finding the Depth First Search of a graph.
    b)Write a program for finding the Breadth First Search of a graph

Reference Books:

1. Fundamentals of Data Structures in C, 2nd edition, Horowitz, Sahani and Anderson-Freed, Universities Press, 2008.

Code: B16 CS 2106

OBJECT ORIENTED PROGRAMMING LAB

Lab : 3 Periods	**Sessionals : 50**
Exam : 3 Hrs.	**Ext. Marks : 50**
	Credits : 2

Course Objectives:

1. To develop programs using basic OOPS concepts such as classes and objects.
2. To implement programs using Inheritance concepts.
3. To implement programs using Exception handling.
4. To develop programs using operator overloading concepts.

Course Outcomes:

1. Student will be able to use OOPs concepts.
2. Ability to apply Inheritance concepts to several problems.
3. Ability to use Exception Handling concepts.

SYLLABUS

Part I: UML

Take a own scenario and draw the UML Diagrams (Structural, Behavioral and Interactive).

Part II: Implement the following programs using C++

1. Write a Program that implements stack operations using classes and objects.
2. Write a Program performing complex number addition using friend functions.
3. Write a Program for complex number addition using operator overloading.
4. Write a Program to perform string operations by overloading operators.
5. Write a Program on hierarchical inheritance showing public, private and protected inheritances.
6. Write a Program for computation of student's result using hybrid inheritance.
7. Write a Program implementing bubble-sort using templates.
8. Write a Program on virtual functions.
9. Write a Program for Templates.
10. Write a Program for copying one file to another file using streams.
11. Write a Program for writing and reading a class object to a file.
12. Write program to implement
    a. One catch block and all Exceptions
    b. using Multiple Catch blocks.
13. Write a program to implement the finally block.
14. Write a program to implement pointers to a derived class and virtual base classes.
15. Write a program to implement conversion of objects between different classes using conversion functions.
16. Write a program to implement function overloading- with various data types, with different number of arguments.

17. Write a program to evaluate mixed mode expressions and implicit type conversions.
18. Write a program to show that there is ambiguity in Multiple Inheritance.
19. Write a program to implement a virtual destructor.
20. Write a program to mimic a bank management system (user logins, requests for withdraw /credit, system verifies whether enough balance is available, update the account summary, etc.)

Part III: Implement the following programs using Java

1. Write a program to give an example for command line arguments.
2. Write a program to sort given list of numbers.
3. Write a program to implement linear search and binary search.
4. Write a program for this operator and super keyword.
5. Write a program that gives demonstration of static variables and methods.
6. Write a program that illustrates the simple inheritance, multilevel inheritance.
7. Write a program that demonstrates the difference between method overloading and overriding.
8. Write a program that demonstrates the difference between method overloading and constructor overloading.
9. Write a program that describes the exception handling mechanism.
10. Write a program that uses try and catch blocks and check whether the given array size is negative or not.
11. Write a program that describes the user defined exception.
12. Write a program that illustrates the creation of threads by using runnable class.
13. Write a program that illustrates the creation of threads
14. Write a program that illustrates the multiple inheritances by using interfaces.
15. Write a program that describes the life cycle of an applet.
16. Write a program that displays the number of characters, lines and words in a text file.
17. Write a program on packages.
18. Write a program to copy contents of a file into another file using File streams.
19. Write a Program for handling Array Index out of Bounds Exception and Divide-by-zero Exception.
20. Write a Program for interfaces.
21. Write a Program on Threads.
22. Write a Program for Constructors.
23. Write a Program for Wrapper Classes.
24. Write a Program for Thread Priority.
25. Write a program for Producer Consumer Problem.

Reference Books:

1. Object Oriented Programming with C++,6e.E.Balagurusamy,TMH,India,2013.
2. Programming with JAVA-A Primer,5e,E.Balagurusamy,TMH,India,2014.
3. The Unified Modeling Language user Guide, Grady Booch, James Rumbaugh and Ivar Jacobson,Addision-wesley.1999.

Code: B16 ENG 2104

ENGLISH PROFICIENCY
(Common to All Branches)

Theory	**: 1 Period**	**Sessionals**	**: 50**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 50**
Exam	**: 3 Hrs.**	**Credits**	**: 2**

AIM:

Enriching the communicative competency of the students by adopting the activity-based as well as the class-oriented instruction with a view to facilitate and enable them to enhance their language proficiency skills.

Course Objectives:

Students be able to

1. Understand the importance of professional communication.
2. Learn language skills and vocabulary in order to improve their language competency.
3. Know and perform well in real life contexts.
4. Identify and examine their self-attributes which require improvement and motivation.
5. Build their confidence and overcome their inhibitions.
6. Improve their strategies in reading skills.

Course Outcomes:

1. Students enhance their vocabulary and use it in the relevant contexts .
2. They improve speaking skills.
3. They learn and practice the skills of composition writing.
4. They enhance their reading and understanding of different texts.
5. They enrich their communication both in formal and informal contexts.
6. They strengthen their confidence in presentation skills.

SYLLABUS

Speaking Skills
PPT
Describing event/place/thing
Picture Description
Extempore
Debate
Telephonic Skills
Analyzing Proverbs

Vocabulary
Affixes
Pairs of Words

Reading Skills
Reading Comprehension
Reading/Summarizing News Paper Artic

Writing Skills
Designing Posters
Essay writing
Resume Writing

Reference Books:

1. Interchange (4th edition) Student's books 1&2 by Jack C. Richards, CUP.
2. Fundamentals of Technical Communication by Meenakshiraman, Sangeta Sharma of OUP
3. English and Communication Skills for Students of Science and Engineering, by S.P.
4. Dhanavel, Orient Blackswan Ltd. 2009
5. Enriching Speaking and Writing Skills, Orient Blackswan Publishers
6. The Oxford Guide to Writing and Speaking by John Seely OUP

(***Note: Sessional Marks will be evaluated based on Continuous Comprehensive Evaluation of the students' Performance - 40M, Attendance – 10M and External Marks will be evaluated based on Presentation Skills – 30M, Project 20M)

Code: B16 ENG 2105

INDUSTRY ORIENTED TRAINING
(WEB Development)
(Common to CSE & IT)

Lab: 2 Periods	**Sessionals : 50**
Exam : 3 Hrs.	**Credits : 1**

Course Objective:

Web development technologies such as HTML, CSS, JavaScript and others are at the core of all modern online systems. The objective of this course is to expose & train the students on web programming concepts and develop the ability to build web sites with dynamic presentation.

Course Outcomes:

Upon completion of this web development training course, able to:

1. Design and develop basic web pages using HTML.
2. Apply cascading style sheets to web pages in order to separate form from content.
3. Understand & Apply basic control of elements with JavaScript.
4. Understand the basic concepts of PHP scripting
5. Able to design & complete a project by applying above all the concepts.

Syllabus: Industry Oriented Applications on following topics.

HTML:- HTML Introduction, HTML Basic Tags, HTML Lists, HTML Tables, HTML Images, HTML Links & Navigation, HTML Forms.

CSS:-CSS Introduction, CSS Properties - Controlling Fonts, CSS Properties - Text Formatting, Selectors - id and class, Pseudoclasses, CSS for Links, CSS for Lists, CSS for Tables.

JAVA SCRIPT:-JavaScript Introduction, Empty Field Validation Example, Name & Numbers Only Validation Example, Email Validation Example, innerHTML Error Display Example.

PHP:-Installation of Wamp Server, PHP Introduction, Creating PHP Script, Running PHP Script, PHP Numeric Variables, Sample PHP Programs

MINI PROJECT

(Note: Total Marks will be evaluated based on Continuous evaluation - 25 Marks, Mini Project- 25 Marks)

SCHEME OF INSTRUCTION & EXAMINATION
(Regulation R16)

II/IV B.TECH
(With effect from **2016-2017** Admitted Batch onwards)
Under Choice Based Credit System

COMPUTER SCIENCE AND ENGINEERING

II-SEMESTER

Code No.	Course	Credits	Lecture Hrs	Tutorial Hrs	Lab Hrs	Total Contact Hrs/Week	Sessional Marks	Exam Marks	Total Marks
B16 CS 2201	Operating Systems	4	3	1	--	4	30	70	100
B16 CS 2202	Computer Organization	4	3	1	--	4	30	70	100
B16 CS 2203	Microprocessors	4	3	1	--	4	30	70	100
B16 CS 2204	Data Communications	4	3	1	--	4	30	70	100
B16 CS 2205	Advanced Data Structures	4	3	1	--	4	30	70	100
B16 CS 2206	Computer Graphics	4	3	1	--	4	30	70	100
B16 CS 2207	Operating Systems & Unix programming Lab	2	--	--	3	3	50	50	100
B16 CS 2208	Digital Electronics & Microprocessors Lab	2	--	--	3	3	50	50	100
B16 CS 2209	Competitive Programming	1	1	--	1	2	50	--	50
B16 ENG 2203	Industry Oriented Training	1	--	--	2	2	50	--	50
	Total	**30**	**19**	**6**	**9**	**34**	**380**	**520**	**900**

Code: B16 CS 2201

OPERATING SYSTEMS

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. To understand operating system as a layer of abstraction above physical hardware that facilitates usage convenience and efficient resource management.
2. To learn design and implementation of policies and mechanisms of OS.
3. To investigate case studies to understand the design paradigms for popular multiuser or single user operating system.

Course Outcomes:

1. The student understands OS evolution, its structure and services provided by it.
2. Learn process life cycle, process scheduling objectives, policies and mechanisms, process synchronization, inter process communication, deadlocks and other process subsystem related concepts.
3. Learn memory hierarchy, allocation, de-allocation policies and mechanism for main and auxiliary memory, file system design and implementation issues.
4. Investigate UNIX/ LINUX and Windows OS platforms w.r.t similarities and differences in design paradigms.

SYLLABUS

Introduction to Operating Systems:
Over View of Operating Systems, Types of Operating Systems, Operating System Structures, Operating System Services, System Calls, Virtual Machines, Operating System Design and Implementation.

Process Management:
Process Concepts, Operations on Processes, Co-operating Processes, Threads, Inter Process Communication, Process Scheduling, Scheduling Algorithms, Multiple - Processor Scheduling, Thread Scheduling.

Process Synchronization:
The Critical Section Problem, Peterson‘s Solution, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors.

Deadlocks:
System Model, Deadlock Characterization, Methods For Handling Deadlocks, Deadlock Prevention, Avoidance, Deadlock Detection, Recovery from Deadlocks

Memory Management:
Logical versus Physical Address, Swapping, contiguous memory allocation, paging, structure of the page table , segmentation, Virtual Memory, Demand Paging, Page Replacement, Allocation of Frames, Thrashing, Memory-Mapped files

File Systems, Implementation, and Secondary-storage Structure:
Concept of a file, Access Methods, Directory Structure, Protection, File System Structure, Allocation Methods, Free Space Management, Directory Management, Device Drivers, overview of Mass-storage structure, Disk structure, disk attachment, disk scheduling, swap-space management.

Case study: Overview of LINUX, Windows Operating systems

Text Book:

1. Operating Systems, Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, John Wiley Publ., Seventh Edition.

Reference Books:

1. Modern Operating Systems, Andrew S. Tanenbaum, , 2nd edition, 1995, PHI.
2. Operating Systems, William Stallings 5th Edition - PHI
3. Operating Systems: A Design-Oriented Approach', Charles Crowley, _Tata Hill Co.,1998 edition.

Code: B16 CS 2202

COMPUTER ORGANIZATION

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. To study about structure and functional components of a computer.
2. Understanding the hierarchical organization of a computer system which consists of Instruction set of commands.
3. Learn about the architecture of a computer.
4. To study about designing a balance system that minimizes performance and utilization of all elements.

Course Outcomes:

1. Apply the basic knowledge about Digital logic to the Functional components of computer.
2. Students will be able to Describe the major components of a computer.
3. Students will be able to classify different Computer Instructions.
4. Students will be able to Describe Instruction set architecture.
5. Recognize the importance of peripheral devices.
6. Students should be able to classify Computer memories

SYLLABUS

Register Transfer and Micro operations:
Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro operations, Logic Micro operations, Shift Micro operations, Arithmetic Logic Shift Unit.

Basic Computer Organization and Design:
Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input- Output and Interrupt, Complete Computer Description, Design of Basic Computer, Design of Accumulator Logic.

Micro programmed Control:
Control Memory, Address Sequencing, Micro program Example, Design of Control Unit.

Central Processing Unit:
Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer(RISC)

Pipeline and Vector Processing:
Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISK Pipeline, Vector Processing, Array Processors.

Input/output Organization:
Peripheral Devices, I/O interface, Asynchronous data transfer, Modes of transfer, priority Interrupt, Direct memory access, Input-Output Processor (IOP), Serial Communication.

Memory Organization:
Memory Hierarchy, Main memory, Auxiliary memory, Associate Memory, Cache Memory, and Virtual memory.

Text Book :

1. Computer System Architecture, M. Morris Mano, Prentice Hall of India Pvt. Ltd., Third Edition, Sept.2008.

Reference Books :

1. Computer Architecture and Organization, William Stallings, PHI Pvt. Ltd., Eastern Economy Edition, Sixth Edition, 2003.
   (Computer Organization and Architecture-Designing of Performance)
2. Computer Organization and Architecture, Linda Null, Julia Lobur, Narosa Publications, Third Edition ,2003.
3. Computer Architecture and Organization, John. P. Hayes, Third Edition, Tata Mc Grawhill International Edition,1998.

Code: B16 CS 2203

MICROPROCESSORS

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. To discuss the architectures of 8085, 8086 microprocessors, their instruction sets and related ALP programs.
2. To discuss interfacing semiconductor memories, interfacing peripheral to Intel 8085.

Course Outcomes:

1. Understand the basic architectures of 8085 and 8086 microprocessors.
2. Ability to write ALP programs using instruction sets of 8085 & 8086.
3. Understand the various interfacing concepts.

SYLLABUS

Internal Architecture functional/signal description of 8085 microprocessor, Instruction set, Addressing modes and programming in 8085.

Timing diagram, counters and time delays, stacks and subroutines and Interrupts in 8085

Classification and interfacing semiconductor memories with 8085 MPU. Interfacing characteristics of IO devices, IO device addressing methods.

Interfacing peripherals to INTEL 8085: Parallel I/O interface-8255, Serial I/O Interface-8251, Timer Interface-8253.

Interfacing peripherals to INTEL 8085: Keyboard/Display Interface-8279, Interrupt controller Interface-8259.

The 8086 Microprocessor architecture: Internal Architecture & functional /signal description of 8086, segmented memory, Maximum & Minimum mode of 8086.

Instruction set and programming the 8086: Addressing modes, Instruction set and assembly language programming techniques with 8086.

Text Books

1. Microprocessor Architecture and Applications with the 8085 , Ramesh S. gaonkar, 4th Edition, Penram International, 1999
2. Advanced Microprocessors and Peripherals, A K RAY & K M Bhurchandi , 2nd Edition,The Mcgraw-Hill companies.

Reference Books

1. The 80X86 Family , Design, Programming and Interfacing, John E. Uffenbeck, 3rd Edition, Pearson Education Inc., 2002.
2. Walter A . tribal and Avatar Singh. The 8088 and 8086 Microprocessors, Programming interfaing, software, hardware and Applications, 4th Edition Pearson education Inc., 2003
3. Microprocessors and Interfacing. Programming and hardware, 2ne Edition, Douglass V. Hall. MH Edition , 1999.

Code: B16 CS 2204

DATA COMMUNICATIONS

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. To study basics of data communication systems.
2. To study the various types of transmission media.
3. To study the various hardware concepts related to data communications.
4. To discuss about multiplexing techniques.

Course Outcomes:

1. Students will have the ability to use Data Communications and Networking Protocols and protocol architectures
2. Students will have the ability to develop communication models for providing data transmission facility
3. Students will have the ability to outline Data Communication terminology
4. Students will have the ability to classify various transmission media
5. Students will have the ability to discriminate various types of signals for data transmission and ability to describe data encoding techniques
6. Students will have the ability to describe data communications interface
7. Students will have the ability to apply various flow control , error control techniques of data link control protocols
8. Students will have the ability to use various data communication terminals and processing hardware
9. Students will have the ability to demonstrate multiplexing techniques

SYLLABUS

Introduction to Data Communications:
A Communications Model, Data Communications and Data Communications Networking, Protocols and Protocol Architecture, Characteristics of Data Transmission: Concepts and Terminology, Analog and Digital Data Transmission, Transmission Impairments.

Transmission Media:
Guided Transmission Media, Wireless Transmission. **Data Encoding:** Digital Data-Digital Signals, Digital Data-Analog Signals, Analog Data-Digital Signals, Analog Data-Analog Signals.

Data Communication Interface:
Asynchronous and Synchronous Transmission, Line Configurations, Interfacing. Data Link Control Flow Control, Error Detection, Error Control, High-Level Data Link Control (HDLC)

Data Communications Hardware:

Terminals: Introduction, Basic Terminal Components, Enhanced Terminal Components, General-Purpose Terminals, Remote Job Entry Terminals, Transaction Terminals, Clustering of Terminal Devices.

Communications Processing Hardware:

Introduction,SwitchingProcessors,MultidropLins,Multiplexers,Concentrators, Front-End Processors

Multiplexing:

Frequency-Division Multiplexing, Synchronous Time-Division Multiplexing: Characteristics, TDM Link Control, Digital Carrier Systems, Statistical Time-Division Multiplexing: Characteristics.

Text Books:

1. William Stallings, Data and Computer Communications, 7th Edition, PH/Pearson Edu.Inc.,
2. Mary E.S. Loomis, Data Communications, PHI-N.J.,1983(Chapter 3, Chapter 5)

.

Reference Books:

1. Behrouz A. Forouzan, Data Communications and Networking, 3 rdEditionTMH,2004
2. William A. Shay, Understanding Data Communications & Networks, 2 ndEdition Thomson-Brooks/Cole –Vikas Publishing House,1999.
3. Michale A. Miller, Data & Network Communications, Thomson/Delmar –Vikas Pub. House, 2000

Code: B16 CS 2205

ADVANCED DATA STRUCTURES

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. To study the concepts related to trees such as binary trees, BST , AVL trees etc.
2. To discuss various hashing technique.
3. To study the various external sorting algorithms.
4. To discuss the concepts related to disjoint set ADT.
5. To study several graph algorithms and their time complexities.

Course Outcomes:

1. Student will be able to write programs to implement various trees.
2. Ability to understand various hashing techniques.
3. Ability to write programs to implement sorting techniques.
4. Ability to understand concepts related to graph theory.

SYLLABUS

Trees:
Definition , operations and applications of Binary search trees, AVL trees, Red-Black Trees, Splay trees, Tries and B-Trees, B+ Trees

Priority Queues:
Heap model and implementations, Binary Heap, Applications of Priority Queues, d-Heaps, Leftist Heaps, Skew Heaps, Binomial Queues structure, operations and Implementation

Hashing & External sorting:
Hash Table Structure, Hash Function, Collision handling, Separate Chaining, Open Addressing, Rehashing, Extendible hashing, Difference between internal and external sorting, Model and simple algorithm for External sorting, Multi-way Merge, Poly-phase Merge, Replacement selection.

Graph algorithms:
Representation of graphs, Topological sort, Network flow problems, Applications of Depth first search for finding Bi-connectivity, Euler circuits, strong components, Introduction of NP-Completeness

Disjoint Set ADT & Amortized analysis:
Equivalence relations, Dynamic equivalence problem, Basic data structure, smart union algorithms, path compression, Analysis of union/find algorithm, applications of ADT Disjoint set, Introduction to amortized analysis, Basic approaches, binary queues, skew heaps, Aggregate analysis, The accounting method, The potential method and Dynamic tables.

Text Books:

1. Data Structures and Algorithm Analysis in C – Mark Allen Weiss, Pearson Edu Publishers.
2. Data Structures and Algorithms: Concepts, Techniques and Applications – G.A.V.Pai, Tata Mc Graw Hill Publishers

Reference Books:

1. Advanced Data Structures – Peter Brass, Cambridge University Press, 2008
2. Introduction to Algorithms by Thomos H.cormen,3 rd Edition, PHI

Code: B16 CS 2206

COMPUTER GRAPHICS

Theory	**: 3 Periods**	**Sessionals**	**: 30**
Tutorial	**: 1 Period**	**Ext. Marks**	**: 70**
Exam	**: 3 Hrs.**	**Credits**	**: 4**

Course Objectives:

1. Provide a comprehensive introduction to computer graphics leading to the ability to understand contemporary terminology, progress and trend.
2. Design algorithms for digitization of graphic primitives
3. Apply computer graphics techniques for two-dimensional and three-dimensional transformations
4. Visualize viewing transformations
5. Demonstrate working of I/O devices

Course Outcomes:

At the end of the course the student will be able to

1. Summarize the application areas of computer graphics
2. Implement algorithms for scan converting graphic primitives in a graphic package.
3. Apply direct and indirect methods for two-dimensional transformations using matrices.
4. Construct three-dimensional geometric transformations using matrices.
5. Visualize two-dimensional viewing transformations
6. Produce views of three-dimensional scenes.
7. Visualize the working of I/O devices

SYLLABUS

Overview of Graphics Systems:
Applications of Computer Graphics-Graphical User Interfaces-Video Display Devices-Raster Scan Systems-Random Scan Systems-Graphics Monitors and Workstations -Input Devices-Logical Classification of Input Devices-Hard Copy Devices- Graphics Software-Overview of C-Graphics, Open GL and PHIGS.

Output Primitives:
Points and Lines-Line Drawing Algorithms- Circle Generating Algorithms- Parallel Line Algorithms-Functions in C-Graphics for Output Primitives.

Attributes of Output Primitives:
Line and Curve Attributes-Color and Gray Scale Levels-Area Fill Attributes-Character Attributes-Bundled Attributes-Inquiry Functions-Anti-aliasing Techniques -Functions in C-Graphics for Attributes of Output Primitives-Filled area primitives- Boundary Fill Algorithm-Flood Fill Algorithm.

Two-Dimensional Geometric Transformations:
Basic Transformations- Matrix Representations-Homogeneous Coordinates-Composite Transformations-Reflection-Shear-Transformations between Coordinate Systems-Affine Transformations- Raster Methods for Transformations.

Two-Dimensional Viewing:
The Viewing Pipeline-Viewing Coordinate Reference Frame-Window-to-Viewport Coordinate Transformation-Clipping Operations-Point Clipping-Line Clipping-Polygon Clipping-Curve Clipping- Text and Exterior Clipping

Three-Dimensional Object Representations and Viewing:
3D Display Methods- 3D Graphics-Polygon Surfaces- Curved Lines and Surfaces- Quadratic Surfaces-Super Quadrics-Blobby Objects-Spline Representations-Cubic Spline methods-Bézier Curves and Surfaces- 3D Viewing Pipeline- Viewing Coordinates- Projections- View Volumes- General Projection Transformations.

Three-Dimensional Geometric Transformations:
Translation- Rotation- Scaling- Reflection – Shear-Composite Transformations-Modeling and Coordinate Transformations.

Case studies- Implementation of algorithms in c-graphics

Text Book:

1. Computer Graphics C Version, Donald Hearn& M. Pauline Baker, Pearson Education, New Delhi, 2004

Reference Books:

1. Procedural Elements for Computer Graphics, David F. Rogers, Tata McGraw Hill Book Company, New Delhi, 2003
2. Computer Graphics: Principles & Practice in C, J.D. Foley, S.K. Feiner, A. Van Dam. F.H John, Pearson Education, 2004
3. Computer Graphics with Open-GL, Donald Hearn, M. Pauline Baker& Warren Carithers,4th Edition, 2011
4. Computer Graphics, Zhigang Xiang and Roy A. Plastock, McGraw-Hill Education, 2nd Edition, ,2015
5. Mathematical and computer programming techniques for computer graphics, Peter Comninos, Springer -Verlag, 2006.

Code: B16 CS 2207

OPERATING SYSTEMS AND UNIX PROGRAMMING LAB

Lab : 3 Periods	**Sessionals : 50**
Exam : 3 Hrs.	**Ext. Marks : 50**
	Credits : 2

Course Objectives:

1. To learn about UNIX/LINUX operating system, its intervals.
2. To learn system programming for UNIX/LINUX Operating System.
3. To understand UNIX/LINUX shell programming.
4. To understand resource management policies, mechanisms and their performance evaluation.

Course Outcomes:

1. The student practices UNIX commands, Vi editor, shell commands.
2. The student develops skill in writing C programs using system calls for process management, inter process communication and memory management aspects.
3. The student learns shell programming and develops skill for writing scripts for batch level tasks.

SYLLABUS

Module I

OS lab familiarization, Home Assignment on Unix commands, Vi editor
Simple C programs using command line arguments, system calls, library function calls, make utility C programs using fork system call to create process and study parent, child process mechanism
C programs to create process chaining, spawning
C programs to handle errors using errno, perror() function
C programs to use pipe system call for inter process communication

Module II

Familiarization of Unix shell programming
Simple shell programming exercises
Shell programming using decision making constructs
Shell programming using loop constructs
Shell programming for file and directory manipulation

Module III

C programs to study process scheduling (FCFS, Shortest Job First, and Round Robin) C programs to study page replacement (FIFO, Optimal, and LRU page replacement) C programs to study deadlock avoidance and detection

C Programs to simulate free space management

Reference Books:

1. Unix concepts and applications by Sumitabha Das, TMH Publications.
2. Unix programming by Stevens, Pearson Education.
3. Shell programming by YashwanthKanetkar.
4. Operating SystemConcepts bySilberschatz, and Peter Galvin

Code: B16 CS 2208

DIGITAL ELECTRONICS AND MICRO PROCESSORS LAB

Lab : 3 Periods	**Sessionals : 50**
Exam : 3 Hrs.	**Ext. Marks : 50**
	Credits : 2

Course Objectives:

1. To learn about logic gates, half adders, full adders and flip -flops.
2. To learn about the microprocessor programming.
3. To learn about the microprocessor interfacing with stepper motor, R-2R ladder.

Course Outcomes:

1. The student understands the logic gates, half adders, full adders and flip-flops to design a circuit.
2. The student develops the skill of writing microprocessor programming.
3. The student understands the interfacing of microprocessor with stepper motor, R-2R ladder.

SYLLABUS

DIGITAL EXPERIMENT

Verification of Truth tables of OR, AND, NOT, NAND, NOR, EX-OR gates(by using 7400-series)

Construction of gates using NAND, NOR gates.

Construction of Half and Full adders and verifying their truth tables.

Operation and verifying truth tables of flip-lops-RS, D and JK using IC's

Up/Down counters using JK flip-flops.

4-bit shift right and left registers using JK flip-flops.

MICROPROCESSORS: 8085

Binary Addition of 'N' 8-bit numbers.

Binary to BCD conversion

Arranging –Ascending/descending order

To find the largest /smallest numbers in the array.

ASCII to HEXA & HEXA to ASCII conversion.

MICROPROCESSORS: 8086

Liner Search

Factorial of a given number

To copy string from S1 to S2

To find GCD and LCM

MICROPROCESSOR INTERFACING WITH 8085

Elevator
Traffic Light.
Analog to Digital & Digital to Analog Conversion
Interrupt controller
Stepper Motor controller.

Reference Books

1. Microprocessor Architecture and Applications with the 8085 , Ramesh S. Gaonkar, 4th-Edition, Penram International, 1999
2. Advanced Microprocessors and Peripherals, A K RAY & K M Bhurchandi , 2nd Edition, The Mcgraw-Hill companies.
3. The 80X86 Family , Design, Programming and Interfacing, John E. Uffenbeck, 3rd Edition, Pearson Education Inc., 2002.
4. Walter A . tribal and Avatar Singh. The 8088 and 8086 Microprocessors, Programming interfacing, software, hardware and Applications, 4th Edition Pearson education Inc., 2003
5. Microprocessors and Interfacing. Programming and hardware, 2ne Edition, Douglass V. Hall. MH Edition , 1999.

Code: B16 CS 2209

COMPETETIVE PROGRAMMING

Theory: 1 Period
Lab : 1 Period **Sessionals : 50**
Exam : 3 Hrs. **Credits : 1**

Course Objectives:

1. To Learn basic concepts of Python Programming Language
2. To Learn various Object Oriented Programming Concepts
3. To Learn various advanced Data structures

Course Outcomes:

At the end of the course the student can able to

1. Write programs using python programming
2. Write algorithms
3. Implement various data Structures
4. To apply object oriented mechanisms
5. To Implement various Advance data Structures like AVL trees, B-Trees, Splay trees etc

SYLLABUS

Introduction to Python

- The basic elements of python
- Branching Programs
- Control Structures
- Strings and Input
- Iteration

Functions, Scoping and Abstraction

- Functions and scoping
- Recursion
- Files

Classes and Object-Oriented Programming

- Abstract Data Types and Classes
- Inheritance
- Encapsulation and Information Hiding

Algorithms and Data structures

- Sequences
- Lists
- Item Ordering
- Two-Dimensional Sequences
- The Minmax

Sets and Maps

- Playing Sudoku
- Sets
- Hashing
- The HashSet Class
- Solving Sudoku

Maps

- Memorization
- Correlating Two Sources of Information

Membership Structures

- Bloom Filters
- The Trie Data type

Balanced Binary Search Trees

- Binary Search Trees
- AVL Trees
- Splay Trees
- Iterative Splaying
- Recursive Splaying

B-Trees

- B-Tree Implementation
- B-Tree Insert
- B-Tree Delete

PROJECT

Reference Books :

1. Kent D. Lee, Steve Hubbard, "Data Structures and Algorithms with Python", Springer Publications.
.

(Note: Total Marks will be evaluated based on Continuous Evaluation - 25 Marks, Coding Contest - 25 Marks)

Code: B16 ENG 2203

INDUSTRY ORIENTED TRAINING
(Common to CSE & IT)

Lab : 2 Periods **Sessionals : 50**
Exam : 3 Hrs. **Credits : 1**

Course Objectives:
Students learn about
1. Linked lists, file operations and their applications.
2. Searching and Sorting algorithms.
3. Stack & Queue operations and their applications.
4. The concepts and applications of Trees and Graphs.

Course Outcomes:
Students will be able to
1. Implement the linked lists in real time applications.
2. Apply the file handling operations.
3. Apply the Searching & Sorting algorithms.
4. Implement Stack & Queue operations.
5. Implement the concepts and applications of Trees and Graphs.

Syllabus: Industry Oriented Applications on following topics.

Linear Linked Data: Singly linked list, operations on a linked list, circular linked list, double linked list, operations on double linked list
Standard Storage: Introduction to files, file types, file modes, file functions

Searching & Sorting: Linear search and Binary search, Bubble sort, Selection sort, Insertion sort,Quick sort, Heap sort, Merge sort: Worst and Average case analysis. Decision Tree Model and(worst case) Lower Bound on Sorting. Sorting in linear time- shell sort, radix sort, bucket sort, counting sort.

Stack & Queue: Stack structure, operations. Stack using linear list data. Stack using linear linked data. Queue structure, operations. Queue using linear list data. Queue using linear linked list. Circular queues.

Non Linear Data: Tree Structure and terminology, Binary Trees, Binary Tree traversals, Applications of Binary Tree, Binary Tree Operations.
Priority queues, union-find sets, (augmented) interval trees, (augmented) balanced BSTs and binary indexed trees, Binary Indexed Tree or Fenwick tree,Segment Tree (RMQ, Range Sum and Lazy Propagation), K-D tree, Union Find Disjoint Set, Tries, Interval Tree

Graphs: Graphs and their basic properties- degree, path, cycle, subgraphs, isomorphism, Eulerian and Hamilton walks, graph coloring, planar graphs, trees. Breadth first search and connected components. Depth first search in directed and undirected graphs.

More Trees: Binary search trees, Operations on BST, balanced binary search trees, AVL trees, Red-Black trees, skip lists, hashing. Priority queues, heaps, Fibonacci heap, union-find, splay trees Interval trees, tries.

(Note: Total Marks will be evaluated based on Continuous Evaluation - 25 Marks, Coding Contest- 25 Marks)