ESTD: 1980

SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE (AUTONOMOUS)
(Affiliated to Andhra University, Visakhapatnam), (Recognized 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

INFORMATION TECHNOLOGY

I-SEMESTER

Code No.	Course	Credits	Lecture Hrs	Tutorial Hrs	Lab Hrs	Total Contact Hrs/Week	Sessional Marks	Exam Marks	Total Marks
B16 IT 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 IT 2102	Object Oriented Programming Using C++	4	3	1	--	4	30	70	100
B16 IT 2103	Digital Logic Design	4	3	1	--	4	30	70	100
B16 ENG 2103	Environmental Studies	2	3	1	--	4	30	70	100
B16 IT 2104	Data Structures Lab.	2	--	--	3	3	50	50	100
B16 IT 2105	Object Oriented Programming Lab Using C++.	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**

Code: B16 IT 2101

DATA STRUCTURES

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

Course Objectives:

1. Assess how the choice of data structures and algorithm design methods impacts the performance of programs.
2. Choose the appropriate data structure and algorithm design method for a specified application.
3. Solve problems using data structures such as linear lists, stacks, queues, binary trees, heaps binary search trees, and graphs and writing programs for these solutions.

Course Outcomes:

1. Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithm.
2. Demonstrate different methods for traversing trees.
3. Compare alternative implementations of data structures with respect to performance.
4. Discuss the computational efficiency of the principal algorithms for sorting and searching

SYLLABUS

Introduction to Data Structures: Review of C Programming, Abstract Data Types, Meaning and Definition of Data Structures, Arrays in C, Recursive Definition and Processes, Recursion in C, Simulation of Recursion, Efficiency of Recursion.

Stacks: Stack as an Abstract Data Type, Primitive Operations, Implementing Stack Operations using Arrays, Infix, Postfix and Prefix: Definitions, Evaluation and Conversions.

Queues: Queue as an Abstract Data Type, Types of Queues, Operations, Implementation using Arrays.

Linked List: Operations, Implementation of Stacks, Queues and priority Queues using Linked Lists, Circular Lists- Insertion, Deletion and Concatenation Operations, Stacks and Queues as Circular Lists, Doubly Linked Lists.

Trees: Binary Trees - Definitions and Operations, Binary Tree Representation- Node Representation, Implicit array Representation, Binary Tree Traversal, Threaded Binary Trees and their Traversal, Trees and their Applications; Tree Searching: Insertion and Deletion of a node from a Binary Search Tree.

Searching: Basic Searching Techniques- Dictionary as an Abstract Data Type, Algorithmic Notation, Sequential Searching and its Efficiency, Binary Search, Interpolation Search.

Sorting: General Background- Efficiency, Asymptotic Notations, Efficiency of Sorting, Bubble Sort and Quick Sort, Efficiency of Quick Sort, Selection Sort, Binary Tree Sort, Heap Sort, Insertion Sort , Shell Sort , Address calculation Sort , Merge and Radix Sort.

Graphs and Their Application: Definition of Graphs, Representation of Graphs, Transitive closure, Warshall's Algorithm, Shortest-Path Algorithm, Linked Representation of Graphs, Topological Ordering of nodes, Graph Traversal and Spanning Forests- Undirected Graphs and their traversals, Minimal Spanning Trees.

Text Books:

1. Data Structures Using C and C++ Yadidyah Langsam, Moshe J. Augennstein and Aaron M. Tanenbaum Prentice Hall of India (2nd Edition).
2. Data Structures, Algorithms and Applications with C++, Sahani Mc-Graw Hill.

Reference Books:

1. Data Structures with C by Seymour lipschutz, Schaum Outline series, 2010.
2. 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 IT 2102

OBJECT ORIENTED PROGRAMMING USING C++

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
4. Understand the concepts of polymorphism

Course Outcomes:

1. Able to outline and describe difference between OOP and POP.
2. Able to recognize and differentiate classes and objects , construct and apply Classes and Objects for real time applications.
3. Able to recognize and construct constructors and destructors and can apply usage of inheritance.
4. Able to understand polymorphism concepts and can apply real time applications.
5. Able to illustrate, identify and apply generic programming concepts with exception handling.

SYLLABUS

Basic Concepts Of OOP: Procedural Paradigms, Object Oriented Paradigm, OOP Principles and Terminology, OOP benefits, Procedure and Object Oriented programming languages, advantages and disadvantages.

Introduction to C++ : Basic Structure C++ Program , variable and Constants, Symbolic Constants , basic data types and derived data type , variable declaration , dynamic initialization, type modifiers, type casting, i/o statements in C++, operators and example programs**,** Control Structures- Programs using all control structures and statements, Functions: Function Prototypes, Function Components, Returning values from functions, actual and formal arguments, parameter passing methods, Inline functions,

Classes and Objects: Introduction to class, class definition, class specification, Member functions, data members, access specifiers, scope resolution operator, Object definition and creation, array of objects, pointers, Pointers to objects, this pointer, dynamic allocation operator, friend functions, const and volatile functions, static members, nested classes, local classes.

Constructors and destructors: Definition of constructor and destructor, default constructor, parameterized constructor, copy constructor, constructor with dynamic allocation, explicit constructor

Inheritance: Definition, base class, derived class, using access specifiers in inheritance, Types of Inheritance, protected data with private inheritance, constructor in derived and base class, abstract classes.

Virtual functions and Polymorphism: Function overloading, arrays and strings, Operator overloading through unary and binary operator, Friend functions, Assignment operator, Stream operator overloading and type conversion; Virtual functions, Pure Virtual function, Dynamic polymorphism, Virtual destructor, Virtual base class, Dynamic casting, Cross casting, Down casting, Program development.

Streams and Files in C++: Stream Classes, Formatted and unformatted data, manipulators, user defined manipulators, file streams, file pointer manipulation; file open and close, file handling, random access, object serialization, name spaces, std namespaces, ANSI string objects and standard template library.

Templates, Exception handling: Class templates, Function templates, Member function templates, Exception handling - try-catch-throw paradigm, exception specification, terminate and un expected functions- uncaught exception, exception handling mechanism, multiple catch, nested try, Rethrowing the exceptions

Text Books:

1. Object Oriented Programming through C++ by Robat Laphore.

Reference Books:

1. Object Oriented Programming in C++: N. Barkakati, PHI
2. Object oriented Programming using C++: E. Balagurusamy, PHI.
3. The Complete reference in C++ by Herbert Shieldt, TMH
4. The C++ Programming Language by B. Stroustrup, Pearson Education

Code: B16 IT 2103

DIGITAL LOGIC DESIGN

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

Course Objectives:

1. To understand Number Systems and conversions.
2. To introduce the basic principles for design of combinational circuit and sequential circuits.
3. To learn simple digital circuits in preparation for computer engineering.
4. To design and learn memory elements and system memory.

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. Octaland 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.StateReductionandAssignment.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 IT 2104

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 operations of a dequeue
7. Write a program for the representation of polynomials using circular 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. Data Structures using C by Tanenbaum

Code: B16 IT 2105

OBJECT ORIENTED PROGRAMMING LAB USING C++

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

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 handling Push On Full and Pop On Empty Exceptions for a Stack.
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.)

Reference Books:

1. Object Oriented Programming in C++: N. Barkakati, PHI
2. Object oriented Programming using C++: E. Balagurusamy, PHI.
3. The Complete reference in C++ by Herbert Shieldt, TMH
4. The C++ Programming Language by B. Stroustrup, Pearson Education

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 improvementand 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 Articles

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

INFORMATION TECHNOLOGY

II-SEMESTER

Code No.	Course	Credits	Lecture Hrs	Tutorial Hrs	Lab Hrs	Total Contact Hrs/Week	Sessional Marks	Exam Marks	Total Marks
B16 IT 2201	Operating Systems	4	3	1	--	4	30	70	100
B16 IT 2202	Computer Organization	4	3	1	--	4	30	70	100
B16 IT 2203	Microprocessors	4	3	1	--	4	30	70	100
B16 IT 2204	Data Communications	4	3	1	--	4	30	70	100
B16 IT 2205	Operations Research	4	3	1	--	4	30	70	100
B16 IT 2206	Java Programming	4	3	1	--	4	30	70	100
B16 IT 2207	Java programming Lab	2	--	--	3	3	50	50	100
B16 IT 2208	Digital Electronics & Microprocessors Lab	2	--	--	3	3	50	50	100
B16 IT 2209	Python 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**

II/IV B.Tech-II Sem
IT

Code: B16 IT 2201

OPERATING SYSTEMS

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

Course Objectives:

1. To understand evolution of Operating System.
2. To understand operating system as a layer of abstraction above physical hardware that facilitates usage convenience and efficient resource management of computer system resources.
3. To learn design and implementation of policies and mechanisms for OS subsystem.
4. To investigate case studies to understand the design philosophies / paradigm 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 and 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 philosophies.

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, Cooperating 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 IT 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
3. instruction set of commands.
4. Learn about the architecture of a computer from a programming view.
5. To design 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 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) 23

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, Memory Management Hardware.

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 IT 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 8086.
3. To study interfacing data converters to 8086
4. To study different programming techniques to implement on Microsoft Assembler.

Course Outcomes:

1. Understand the basic architectures of 8085 and 8086 microprocessors.
2. Ability to write ALP using instruction sets of 8085.
3. Understand the various interfacing concepts.
4. Ability to write ALP using instruction sets of 8086.
5. Understand how to interface peripherals with 8086.

SYLLABUS

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

Timing diagram, counters and 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 IO interface-8255, Serial IO 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 7 Minimum mode of 8086.

Introduction 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 interfacing, 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 IT 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 modem and 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.

Communication 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, 2nd Edition Thomson-Brooks/Cole –Vikas Publishing House,1999.
3. Michale A. Miller, Data & Network Communications, Thomson/Delmar –Vikas Pub. House, 2000

Code: B16 IT 2205

OPERATIONS RESEARCH

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

Course Objectives

1. To discuss about basic Operation Research concepts , Formulation of LPP and its solution using graphical method.
2. To discuss about standard form of LPP. solving LPP using various methods.
3. To study the various solutions of transportation problems and assignment problems.
4. To discuss about PERT and CPM charts
5. To discuss about replacement problems, inventory problems and game theory.

Course Outcomes:

1. Ability to solve LPP problems using various methods.
2. Ability to solve transportation and assignment problems using several methods.
3. Analyze the PERT and CPM charts
4. Ability to solve replacement problems and game theory problems.

Overview of Operations Research, Types of OR Models , Phases of Operations Research– OR Techniques, Introduction to Linear Programming, Formulation of Linear Programming Problem, Graphical Solution; Graphical Sensitivity Analysis,

Standard Form of LPP, Basic Feasible Solutions, Unrestricted Variables, Simplex Algorithm , Artificial Variables, Big M Method, Two Phase Simplex Method, Degeneracy, Alternative Optimal, Unbounded Solutions, Infeasible Solutions, Primal And Dual Problems And Their Relations, Dual Simplex Method

Transportation Problem as LPP, Initial Solutions, North West Corner Rule, Lowest Cost Method, Vogels Approximation Method, Optimum Solutions of TPP, Degeneracy in Transportation, Transportation Algorithms,

Assignment Problem, Assignment Problem as LPP, Hungarian Method, Travelling Salesman Problem, Solutions of TSP, Sequencing Problems, N-Jobs Two Machine Problems, N-Jobs K Machines Problems, Two-Jobs M- Machine Problems, Crew Scheduling Problems

Network Representation of A Project, CPM and PERT, Critical Path Calculations, Time – Cost Optimizations, PERT Analysis and Probability Considerations, Resource Analysis in Network Scheduling.

Inventory Control- Inventory-Factors Effecting Inventory-EOQ, Inventory Problems With and Without Shortages, Inventory Problems with Price Breakups, Multi Item Deterministic Problems. Probabilistic Inventory Problems

Game Theory: Two Person Zero Sum Games, Mixed Strategy Games and Their Algorithms.

Text Books:

1. Operations Research, Kanti Swaroop, P.K. Gupta, Man Mohan, Sulthan Chand& Sons Education
2. Publishers Operations Research – An Introduction, Handy A Taha – Pearson Education .

Reference Books:

1. Operations Research Panneer Selvan Prentice Hall Of India.
2. Operations Research By S.D Sharma
3. Introduction To Operations Research, F.S. Hiller, G.J. Liberman, TMH
4. Operations Research, Richard Bronson, Schaum‘s Series

Code: B16 IT 2206

JAVA PROGRAMMING

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

Course Objectives:

1. To know basic principles of Object Oriented Programming in the context of java programming language.
2. To study different types of arrays to design programs.
3. To study and apply different types of java concepts like multithreading ,packages ,exception handling ,interfaces and design programs using these concepts.
4. To study and apply various features of AWT components.
5. To know the basic concepts of networking in the context of java programming.

Course Outcomes:

1. Ability to define different procedural and object oriented concepts and will be able to apply and differentiate between them.
2. Ability to define, understand and differentiate different types of arrays and apply them**.**
3. Ability to recognize various concepts of java and develops the programs using them.
4. Ability to identify and differentiate the various features of AWT components to construct container based programs.
5. Ability to describe and explain the concept of networking.

SYLLABUS

Fundamentals: HTML, OOP Concepts, Comparing JAVA with C & C++,JAVA Programming language Syntax, Variables, Data types, statements and expressions.

Control Statements: If else, for, while, and do while loops, Switch statements.

Arrays & Structures: One Dimensional & Two Dimensional Arrays, Named Structures.

Functions: Parameter Passing, Static Modifier.

Features of JAVA: Classes and Interfaces, Threads and multithreaded programming, Exception handling, Introduction to packages, Math package, Lang package, Util package.

Applet Programming: Events, Event driven programming, Events like buttons, mouse, keyboards etc., Applets, Applets package, Fonts, colours, Graphics, images. AWT components, layout managers, writing event driven program using components.

Networking:

Networking Basics: Socket overview, Client/Server, Reserved sockets. Proxy servers, Internet addressing; Java and the net, Inet address, TCP/IP client sockets, URL, URL connection, TCP/IP server sockets, Datagrams.

Textbooks:

1. Introduction to Java programming, a primar ǁ, Balaguruswamy.
2. Java Complete Reference , Herbt Schild.

Reference Book:

1. Introduction to Java programming, Daneal/Young PHI

Code: B16 IT 2207

JAVA PROGRAMMING LAB

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

Course Objectives:

1. To demonstrate the compilation and interpretation of java programs
2. To make practice on different Objet Oriented Programming Concepts
3. To write java programs to implement java concepts
4. To write java programs to implement applet programming
5. To write java programs to generate and handle event handling programs

Course Outcomes:

1. Students will be able to understand compiling and interpreting programs.
2. Students will be able to Explore features of Object Oriented Programming.
3. Students will be able to implement various java concepts
4. Students will be able to Develop java Programs to implement applets
5. Students will be able to Develop java Programs to generate and handle events.

LIST OF PROGRAMS

1. (a) Program to display the area of a rectangle.
   (b) Program to find Sum of series $1+x+x^2+x^3+\ldots\ldots$
2. (a) Write a class to display the area of rectangle and inherit this class into other class which is displaying perimeter of a rectangle and implement.
   (b) Write a class to add three no‘s inherit this class into other class to add five no‘s and implement it.
3. (a) Write a program to print the path, filename and extension for a given path of a file.
   (b) Write a program to receive two command line arguments check whether they are equal or not.
4. (a) A program to take two arguments and divide the first argument with second argument and display the result. Displays the error message if divide by zero without abnormal exit.
   (b) A program to accept more than one string and arrange them in alphabetical order.
   (c) Write a program to display simultaneously output of even and odd numbers starting from one to specified number.
5. Write a program to accept data from keyboard and write it into a file.
6. Write a java program to implement stack & Queue operations.
7. Write a program to draw line and circle using mouse.
8. Write an applet program for drawing the bar chart.
9. Write an applet program to design a calculator for implementing basic functions like +,-,*, /.
10. Write a program to check active ports in system.

Reference Books:

1. Introduction to Java programming, a primar Ⅱ, Balaguruswamy.
2. Java Complete Reference , Herbt Schild.
3. Introduction to Java programming, Daneal/Young PHI

Code: B16 IT 2208

DIGITAL ELECTRONICS AND MICROPROCESSORS 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, elevator.
4. To learn about the Microprocessor interfacing with R-2R ladder network.

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 with 8085.
3. The student understands the interfacing of microprocessor with stepper motor, R-2R ladder.
4. The student will be able to write ACP for 8086.

LIST OF PROGRAMS

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 LCD

MICROPROCESSOR INTERFACING WITH 8085

Elevator
Traffic Light.
Analog to Digital & Digital to Analog Convertors
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 interfaing, 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 IT 2209

PYTHON 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

Overview, Environment Set Up, Basic Syntax, Identifiers, Reserved Words, Lines and Indentation, Multi-Line Statements, Quotation, Comments, Multiple Statements on a Single Line Variable Types, Standard Data Types, Numbers (math, random, fraction) , Strings, Lists, Tuples , Dictionaries

Operators, Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Decision Making :if, if-else, nested if , Loops: for, while, nested loops

Functions, Function Arguments: Required arguments, Keyword arguments, Default arguments, Variable-length arguments, The Anonymous Functions: lambda, Scope of Variables, Modules, sys, os , Date & Time

Files & its operations, Exceptions, Standard Exceptions, Assertions, The try-finally Clause, Raising an Exception, User-Defined Exceptions, Classes and objects , OOPS, Data member , Function overloading, Instance variable, Inheritance, Instance, Instantiation, Operator overloading.

HTML,CSS Basics, Data Base(SQLite), Database Connection, CRUD Application , CGI Architecture, Web Server Support and Configuration, GET and POST Methods, CGI Scripts

Project Work

Text Books:

1. Python Cookbook, D. Beazkey & B.K. Jones, O'Reilly Series, 3rd Edition,
2. Python Programming for beginner " The comprehensive guide to Python Programming ", 1st Edition, Adam Stark

Reference Books :

1. Kent D. Lee, Steve Hubbard, "Data Structures and Algorithms with Python", Springer Publications. Prabhanjan Narayanachar Tattar, Suresh Ramaiah, B.G. Manjunath, "A course in statistics with R", WILEY 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)