NAGARJUNA

COLLEGE OF ENGINEERING & TECHNOLOGY

An Autonomous College under VTU

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

VISION

To disseminate the IT knowledge among the students for achieving excellence in education and to irradiate budding engineers as leaders in information technology

MISSION

M1. To maintain leadership and excellence in Information Technology.

M2. Achieving excellence in IT through analysis, design, development of software products.

M3. Developing communication skills, leadership qualities and team work among students community by providing opportunities to work on various projects through internship with industry partners.

M4. To inculcate Ethics and Human values for solving societal problems and environmental protection.

M5. Promoting research, higher studies and entrepreneurship among the students through outside world interaction.

III & IV Semesters

Scheme and Syllabus

With effect from Academic Year 2017 -18

Third Semester B.E. – Scheme

Sl. No.	Course Code	Course	Teaching Dept.	L-T-P-S (Hrs/week)	Total Credits	Marks
1	16ISM31	Engineering Mathematics-III (IC)	Mathematics	3-0-2-0	4	100
2	16IST32	Fundamentals of Computation Engineering	ISE	3-0-0-0	3	100
3	16ISI33	Data Structures with C (IC)	ISE	3-0-2-4	5	100
4	16IST34	Analog and Digital Electronics	ISE	3-0-0-0	3	100
5	16IST35	Computer Organization	ISE	3-0-0-0	3	100
6	16ISI36X	Foundation Elective-I (IC)	ISE	2-0-2-0	3	100
7	16ISL37	Analog and Digital Electronics Laboratory	ISE	1-0-2-0	2	100
8	16ISI38	Virtualization Foundations (IC)	ISE	1-0-2-0	2	100
9	16ISH39	Integrated Rural Development – Part 1	ISE	0-2-0-0	1	100
TOTAL				**19-2-10-4**	**26**	**900**

Foundation Elective-I (IC)

Sl. No.	Course Code	Course
1	16ISI361	Computer Communication and Networking
2	16ISI362	Creating Interactive and Responsive Web Pages
3	16ISI363	Principles of Programming

Fourth Semester B.E - Scheme

Sl. No.	Course Code	Course	Teaching Dept.	L-T-P-S (Hrs/week)	Total Credits	Marks
1	16ISM41	Engineering Mathemaics-IV (IC)	Mathematics	3-0-2-0	4	100
2	16IST42	Formal Languages and Automata Theory	ISE	3-0-0-0	3	100
3	16IST43	Design and Analysis of Algorithms	ISE	3-0-0-0	3	100
4	16ISI44	UNIX and Shell Programming (IC)	ISE	3-0-2-0	4	100
5	16ISI45X	Foundation Elective-II (IC)	ISE	3-0-2-0	4	100
6	16IST46X	Engineering Elective-III	ISE	3-0-0-0	3	100
7	16ISL47	Design and Analysis of Algorithms Laboratory	ISE	1-0-2-0	2	100
8	16ISI48	Cloud Computing Foundations (IC)	ISE	1-0-2-0	2	100
9	16ISH49	Integrated Rural Development – Part 2	ISE	0-2-0-0	1	100
TOTAL				**20-2-10-0**	**26**	**900**

Foundation Elective-II (IC)

Sl. No.	Course Code	Course
1	16ISI451	Introduction to Microprocessors
2	16ISI452	Object Oriented Programming with C++
3	16ISI453	Introduction to Programming using Python

Engineering Elective-III

Sl. No.	Course Code	Course
1	16IST461	Cyber Security
2	16IST462	Renewable Energy Resources
3	16IST463	Smart Materials

IC – Integrated Course **L – Lecture** **T-Tutorials**
P-Practical **S – Self Study**

Engineering Mathematics-III (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISM31	3:0:2:0	4	CIE:50 SEE:50	3 Hours	BS

Course Objectives:

This course will enable students to :

- The course is aimed at developing the application of mathematical skills in solving the engineering problems using computers.
- Learn to use the partial differential equations in engineering applications.
- Use of Transforms in the engineering problems.
- Able to find the approximated solutions to engineering problems numerically.

Syllabus

Module - I

Partial Differential Equations: Formation of PDE –Eliminating the Arbitrary constants and arbitrary functions, solutions of non homogenous PDE by direct integration., Method of separation of variables. Applications to PDE –Derivation of one dimensional of wave equation and solution by separation of variables-with specified boundary conditions. Derivation of one dimensional of Heat equation and solution by separation of variables-with specified boundary conditions. **08 Hours**

Module - II

Fourier Series: Periodic functions, Dirchlet's conditions, Euler's Formulae-Fourier series of periodic functions of period 2l and 2π, Half range Fourier series, Practical harmonic analysis. **08 Hours**

Module - III

Numerical Methods-I: Numerical solutions of Algebraic and transcendental equations-Regula Falsi Method and Newton Raphson Method. Finite Differences-Forward, Backward and Central differences, Newton's Forward Newton's Backward and Sterling's interpolation formulae. Lagrange's Interpolation formula (without proof). Numerical Differentiation using Newton's Forward and Backward formulae.

08 Hours

Module - IV

Numerical Methods-II: Numerical Integration-Trapezoidal rule, Simpson's $1/3^{rd}$ and $3/8^{th}$ rule. Numerical solutions of ordinary differential equations of first order and first degree- Picard's method, Taylor's Series method , Modified Eulers method Runge-Kutta Method of 4^{th} order and Milne's Predictor Corrector Method. **08 Hours**

Module - V

Introduction to SCILAB and its family, Menus and toolbars, Types of windows and types of files, SCILAB Help system, Basic calculations in SCILAB, Basic variables, Functions- Elementary Mathematical, Builtin and User defined functions. Array operations, Matrix operations, Loops: for and while loops, condition statements- if-then and if-then-else statements, plotting of graphs, working with scripts and files.

08 Hours

Sl. No.	Name of the Experiment
1	SCILAB Environment
2	Basic operations in SCILAB
3	Basic Matrix operations
4	SCILAB programming environment
5	Use of Functions
6	Plotting of 2D and 3D Curves
7	Polynomial Evaluation and Determination of Roots of a Polynomial
8	Statistics Using SCILAB
9	Differentiation and Integration using SCILAB
10	Numerical Methods using SCILAB

Course Outcomes:

On completion of this course, the students are able to :

- Form a partial differential equations and their solutions.
- Expressing the given functions as infinite series of sine and cosine.
- Find approximated solutions by numerical methods.
- Use the SCILAB to solve the various types engineering problems.

Text Books:

1. Dr. B.S. Grewal: “Higher Engineering Mathematics”, (Chapters 10, 17, 18, 22, 23, 28-30), Khanna Publishers, New Delhi, 42nd Edition, 2012, ISBN 13: 9788174091956.
2. N.P. Bali and Dr. Manish Goyal: “A Text Book of Engineering Mathematics”, (Chapters 10, 16, 17, 20, 22, 23), Laxmi Publications (P) Ltd., New Delhi, 9th Edition, 2014, ISBN: 9788131808320.
3. SCILAB Group: “Introduction to SCILAB, A Users Guide”.

Reference Books:

1. Erwin Kreyszig: "Advanced Engineering Mathematics", (Chapters 11,12,19), Wiley Pvt. Ltd India, New Delhi, 9th Edition, 2011, ISBN 13: 9788126531356.
2. B.V. Ramana: "Higher Engineering Mathematics", (Chapters 17-21,32), Tata Mc Graw – Hill Publishing company Limited, New Delhi, 2nd Reprint, 2007, ISBN 13: 978-0-07063417-0.
3. S.S. Sastry: "Introductory methods of numerical analysis", (Chapters 2,3,6), PHI Learning Private, Delhi, 5th Edition, 2013, ISBN: 978-81-203-4592-8.
4. Stormy Attaway: "A practical introduction to programming and problem solving", Elsevier, Boston, 2nd Edition.

E-Resources:

1. http://bookboon.com/en/essential-engineering-mathematics-ebook.
2. https://www.free-ebooks.net/ebook/essential-engineering-mathematics.
3. https://www.scilab.org/resources/documentation/books
4. https://archive.org/details/AdvancedEngineeringMathematics10thEdition
5. https://mars.uta.edu/mae3183/simulation/introscilab_baudin.pdf

Fundamentals of Computation Engineering

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16IST32	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand the logical notation of fundamental concepts such as sets, relations and functions.
- Understand the syntax and semantics of propositional and predicate logic.
- Translate statements from a natural language into its symbolic structures in logic.
- Understand the basic concepts of graph theory.
- Learn how to use graphs as a powerful modeling tool to solve practical problems in various fields.
- Get familiarized with modeling of computational methods.

Module - I

Set Theory: Sets and Subsets, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams.

Fundamentals of Logic: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic. **08 Hours**

Module - II

Fundamentals of Logic (Contd.): Rules of Inference, Quantifiers, Definitions and Proofs of Theorems.

Relations: Cartesian Products and Relations, Properties of relations, Equivalence and Partial order relations. **08 Hours**

Module – III

Relations (Contd.): Relations as matrices and graphs, Posets and Hasse diagrams.

Functions: Functions -Plain and One-to-One, Onto Functions –Special Functions, Pigeonhole Principle, Function Composition and Inverse Functions. **08 Hours**

Module – IV

Introduction to Graph Theory: Definitions and Examples, Subgraphs, Complements, and Graph Isomorphism, Vertex Degree, Euler Trails and Circuits, , Hamilton Paths and Cycles. Graph Coloring, and Chromatic Polynomials.

Trees: Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes. **08 Hours**

Module – V

Fundamental Principles of Counting: The Rules of Sum and Product, Permutations, combinations – The Binomial Theorem, Combinations with Repetition.

The Principle of Inclusion and Exclusion: The Principle of Inclusion and Exclusion, Generalizations of the Principle. **08 Hours**

Course Outcomes:

On completion of this course, the students are able to :

- Use logical notation to define and reason about fundamental mathematical concepts such as sets, relations and functions and develop the syntax and semantics of propositional and predicate logic.
- Prove and disprove results related to logic and define, compare and recognize relations.
- Define, compare and recognize functions and identify relations and functions with graphs, tables and sets of ordered pairs.
- Apply the abstract concepts of graph theory in modeling and solving non-trivial problems in different fields of study.
- Demonstrate the ability to solve problems using counting techniques and combinatorics.

Text Book:

1. Ralph P. Grimaldi: "Discrete and Combinatorial Mathematics", (Chapters 1-3, 5, 7, 8, 11, 12), Pearson Education, 5th Edition, 2006, ISBN: 8177584243, 9788177584240.

Reference Books:

1. Kenneth H. Rosen: "Discrete Mathematics and its Applications", McGraw Hill, New Delhi, 7th Edition, 2010, ISBN : 0073383090.
2. J K Sharma: "Discrete Mathematics", Trinity, India, 4th Edition, 2015, ISBN: 978-93-5138-143-3.
3. D.S. Chandrasekharaiah: "Graph Theory and Combinatorics", Prism, Bengaluru, 4th Edition, 2013, ISBN: 978-81-7286-698-3.
4. Richard A. Brualdi: "Introductory Combinatorics", Pearson Education, India, 4th Edition, 2004, ISBN: 978-0-13-602040-0.

E-Resources:

1. https://www.pearsoned.co.in/grimaldidcm5e

Data Structures with C (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI33	3:0:2:4	5	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand common data structures and be able to implement them.
- Understand realization of fundamental data structures like stacks, queues, link list, trees.
- Differentiate the costs and benefits of dynamic and static data structure implementations.

Syllabus

Module - I

Basic Concepts: Basic Concepts: Pointers and Dynamic Memory Allocation, Arrays **and Structures:** Arrays, Dynamically Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices. **07 Hours**

Module - II

Stacks and Queues: Stacks, Stacks using Dynamic Arrays, Queues, Circular Queues using Dynamic Arrays, Evaluation of Expressions, Multiple Stacks and Queues. **09 Hours**

Module - III

Linked Lists: Singly Linked lists and Chains, Representing Chains in C, Linked Stacks and Queues, Polynomials, Additional List operations, Sparse Matrices, Doubly Linked Lists. **08 Hours**

Module - IV

Trees: Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees, Heaps, Binary Search Trees. **08 Hours**

Module - V

Graphs: The Graph Abstract Data Type, Elementary Graph Operations. Priority **Queues:** Single and Double Ended Priority Queue; Leftist Trees Efficient Binary Search **Trees:** AVL Trees, Red-Black Trees. **08 Hours**

List of Experiments:

1. Write a C program to convert and print a given valid parenthesized infix arithmetic expression to postfix expression. The expression consists of single character operands and the binary operators + (plus), - (minus), * (multiply) and / (divide) using virtual lab.
2. Write a C program to evaluate a valid suffix/postfix expression using stack. Assume that the suffix/postfix expression is read as a single line consisting of non-negative single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), * (multiply) and / (divide).
3. Write a C program to simulate the working of a queue of integers using an array. Provide the following operations:

   a. Insert b. Delete c. Display
4. Write a C program to support the following operations on a doubly linked list where each node consists of integers:

   a. Create a doubly linked list by adding each node at the front.

   b. Insert a new node to the left of the node whose key value is read as an input

   c. Delete the node of a given data, if it is found, otherwise display appropriate message.

   d. Display the contents of the list.

(**Note**: Only either (a,b and d) or (a, c and d) may be asked in the examination).

5. Write a program in C to create a max heap of integers by accepting one element at a time and by inserting it immediately in to the heap. Use the array representation for the heap. Display the array at the end of insertion phase.
6. Write a C Program to construct a binary search tree of integers to perform the operations - insert, delete and search using virtual lab.
7. Write a program in C to create a Binary Tree, with functions to perform inorder, preorder and postorder traversals.
8. Write a C program for the construction of AVL tree.

Course Outcomes:

On completion of this course, the students are able to :

- Identify the purposes of dynamic memory allocations.
- Implement the concept of stacks and queues for different problems.
- Apply data set operations using different types of linked list.

- Develop applications to solve tree oriented problems.
- Develop solutions for problems based on graphs.

Text Book:

1. Horowitz, Sahni, Anderson-Freed: "Fundamentals of Data Structures in C", 2nd Edition, Universities Press, 2011, ISBN-13: 978-0716780427 (Chapters 1.2, 2.1 to 2.5, 3.1 to 3.4, 4.1 to 4.8, 5.1 to 5.7, 6.1 to 6.2, 9.1 to 9.2, 10.1 to 10.3).

Reference Books:

1. Yedidyah, Augenstein, Tannenbaum: "Data Structures Using C and C++", 2nd Edition, Pearson Education, 2009, ISBN-13: 9788120311770.
2. Richard F. Gilberg and Behrouz A. Forouzan: "Data Structures A Pseudocode Approach with C", Cengage Learning, ISBN-13: 978-0534390808.

E-Resources:

1. http://www.facweb.iitkgp.ernet.in/isg/PDS/SLIDES/L1-1.pdf.
2. http://www.docs-engine.com/pdf/1/data-structure-using-c-by-tanenbaum.html.
3. http://cse01-iiith.vlabs.ac.in

Analog and Digital Electronics

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16IST34	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand diode as clippers and clampers. Q-point calculation for a given transistor-biased circuit and transistor AC models.
- Learn the design of Non linear op-amp and 555 timer circuits.
- Familiarize VHDL concepts and logic circuits.
- Understand the concepts of Read-only memories; Programmable logic arrays (PLAs). Programmable array logic (PALs), other sequential programmable logic devices (PLDs).

Syllabus

Module - I

Diode Circuits: Basic Ideas, the Ideal Diode, Clippers and Limiters, Clampers, Optoelectronic Devices.

Transistor fundamentals and AC Models: Operating point, The Load line, Transistor Switch, Base-Biased Amplifier, Emitter-Biased Amplifier, Small-Signal Operation.

MOSFETs: The Depletion-Mode MOSFET, D-MOSFET Curves, Depletion-Mode MOSFET Amplifiers, The Enhancement-Mode MOSFET, The Ohmic Region, Digital Switching, CMOS. **08 Hours**

Module - II

Nonlinear Op-Amp Circuits: Comparators with Zero Reference, Comparators with Nonzero References, Comparators with Hysteresis, Window Comparator, the Integrator, Waveform Conversion, Waveform Generation.

Oscillators: The 555 Timers, Astable Operation of the 555 Timer, 555 Circuits.

08 Hours

Module - III

Digital Logic: Overview of Basic Gates and Universal Logic Gates, AND-OR-Invert Gates, Positive and Negative Logic, Introduction to HDL. Combinational Logic Circuits: Boolean Laws and Theorems, Sum of Products Method, Truth Table to Karnaugh Map, Pairs, Quads and Octets, Karnaugh simplifications, Don't Care Conditions, Product-of-sums method, Product-of-sums simplification, Simplification by Quine-McClusky Method, Hazards and Hazard cover. **08 Hours**

Module - IV

Flip-Flops: RS Flip-flops, Gated Flip-flops, Edge-triggered RS, D, JK Flip-flops, Flip-flop timing, JK Master-slave Flip-flops, Switch Contact Bounce Circuits, Various representations of Flip-flops. **08 Hours**

Module - V

Counters: Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus, Decade Counters, Counter Design as a Synthesis Problem, Counter Design Using HDL. **08 Hours**

Course Outcomes:

On completion of this course, the students are able to :

- Design and develop different electronic circuits used in real time applications.
- Analyze and identify diodes, transistors, MOSFET, comparator and 555 timer circuits and their working principles.
- Design of logic circuit using VHDL.
- Distinguish Read-only memories, Programmable Logic Arrays (PLAs), Programmable Array Logic (PALs).
- Design counters and implement the register applications.

Text Books:

1. Albert Malvino, David Bates: "Electronic Principles", 8th Edition, TMH Publications, 2015, ISBN 13: 978-0073373881, (Chapters 3.1 to 3.2, 4.10 to 4.11, 5.8, 7.2, 7.3, 7.5, 9.1 to 9.3, 14.1 to 14.7, 22.1 to 22.3).
2. Donald P Leach, Albert Paul, Malvino and Gautham Saha: "Digital Principles and Applications", 7th Edition, TMH, 2010, ISBN 13: 9780070141704, (Chapters 2,3,4, 8,9,10).

Reference Books:

1. Anil K Maini, Varsha Agarwal: "Electronic Devices and Circuits", Wiley, 2009, ISBN 13: 8174091297.
2. Charles H. Roth Jr.: "Fundamentals of Logic Design", 5th Edition, Thomson, 2010, ISBN-13: 9781133628477.

E-Resources:

1. http://www.electronics-tutorials.ws
2. http://www.circuitstoday.com/category/basic-electronics
3. http://elearning.vtu.ac.in

Computer Organisation

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16IST35	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- The basic structure and operation of a digital computer and performance issues.
- The different ways of communicating with I/O devices and standard I/O interfaces.
- The details of the arithmetic operations including the algorithms and implementation.
- The hierarchical memory system including cache memories and virtual memory.

Syllabus

Module - I

Basic Structures of Computers: Computer Types, Functional Units, Basic Operational Concepts, Bus Structures, Software, Performance, Multiprocessors and Multicomputer, Historical Perspective. Machine instructions and Programs: Numbers, Arithmetic Operations and Characters, Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing. **08 Hours**

Module - II

Machine instructions and Programs: Addressing Modes, Assembly Language, Basic Input Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Example Programs, Encoding of Machine Instructions. **08 Hours**

Module - III

Input/output Organization: Accessing I/O Devices, Interrupts, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces: PCI and SCSI. **08 Hours**

Module - IV

The Memory System: Some Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed Size and Cost, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements, Secondary Storage.

08 Hours

Module - V

Arithmetic: Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-Operand Multiplication, Fast Multiplication, Integer Division, Floating Point Numbers and Operations, Implementing Floating Point Operations. **08 Hours**

Course Outcomes:

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

- Identify functional units, bus structure and performance measures of computer system.
- Demonstrate programming proficiency using the various addressing modes and data transfer instructions.
- Apply the knowledge in identifying the I/O devices and related interfacing circuits.
- Identify memory hierarchy and evaluate memory performance.
- Design Arithmetic Logic Unit using various algorithms.

Text Book:

1. Carl Hamacher, Zvonko Varnesic, Softwat Zaky: "Computer Organization and Embedded systems", (Chapters 1,2,4-6), Mc-Graw Hill Publishers, 6th Edition, 2011, ISBN 13: 9780071089005, ISBN 10: 0071089004.

Reference Books:

1. William Stallings: "Computer Organisation and Architucture", Pearson, 8th Edition, 2010, ISBN-13: 978-8131732458.
2. Vincent P. Heuring: "Computer Systems Design and Architecture", Pearson Education, 2nd Edition, 2008, ISBN-13: 978-8177584837.
3. M. Morris Mano: "Computer System Architecture", Pearson Education, 3rd Edition, 2007, ISBN-13: 978-8131700709.

E-Resources:

1. http://www.faadooengineers.com/threads/80-Computer-Organization-Notes-and-study-material-Full-Course
2. https://www.smartzworld.com/notes/computer-organization-co/
3. http://www.srmuniv.ac.in/downloads/computer_architecture.pdf

Computer Communication and Networking (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI361	2:0:2:0	3	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Understand the ISO networks layers and design.
- Identify the need and techniques for data transitioning from Digital to Digital, Analog to Digital, Digital to Analog and Analog to Analog.
- Get exposed to different Switching techniques.
- Analyze different techniques to improve Error detection methods.

Syllabus

Module - I

Introduction to Communications: Data Communications, Networks, Network Types, Network Models – Protocol layering, TCP/IP Protocol Suite, The OSI Model, Physical layer: Transmission impairment, Date Rate Limits, Performance. **05 Hours**

Module - II

Digital Transmissions 1: Digital to Digital Conversion, Analog to Digital Conversion, Transmission Modes. **06 Hours**

Module - III

Digital Transmissions 2: Analog Transmission- Digital to Analog conversion, Analog to Analog conversion. Multiplexing and Spectrum Spreading-Multiplexing, Spread Spectrum. **05 Hours**

Module - IV

Switching 1: Introduction, Circuit switched networks, Packet Switching, Structure of a Switch. Error Detection and Correction-Introduction, Block Coding. **05 Hours**

Module - V

Switching 2: Cyclic Codes – CRC, Polynomials, Cyclic code encoder using Polynomials, Cyclic code analysis, Advantages of cyclic codes. Checksum, Forward Error Correction.

05 Hours

List of Experiments:

1. Write a program to convert digital to analog data transmission.
2. Write a program to convert analog to digital data transmission.
3. Write a program for error detecting code using CRC-CCITT (16 bits).
4. Using TCP/IP sockets write a client server program to make the client send the file name and to make the server send back the contents of the requested file if present.

Course Outcomes:

On completion of this course, the students are able to :

- Describe the ISO networks layers and design.
- Differentiate between Digital to Digital, Analog to Digital, Digital to Analog and Analog to Analog.
- Apply the knowledge of multiplexing skills to solve the transmission problems.
- Solve problems of different Switching techniques.
- Analyze and implement the CRC and Error detection methods.

Text Book:

1. Behrouz A. Forouzan: "Data Communication and Networking", McGraw Hill, 5th Edition, Copyright: 2013, (Chapters 1-8,10). ISBN-13: 978-0073376226, ISBN-10: 0073376221.

Reference Books:

1. William Stallings: "Data and Computer Communication", 8th Edition, Pearson Education, 2007, ISBN-13: 978-0-13-607373-4 , ISBN-10: 0-13-607373-5.
2. Craig Zacker : "The Complete Reference Networking", McGraw-Hill, 2015.
3. Wayne Tomasi: "Introduction to Data Communications and Networking", Pearson Education, 2013.

E-Resources:

- http://www.mhhe.com/engcs/compsci/forouzan/frontmatter.pdf
- http://ebookinga.com/data-communication-and-networking-tata-mcgraw-hil...
- https://www.goodreads.com/book/show/209441.Introduction_to_Data_Communications_and_Networking

Creating Interactive and Responsive Web Pages (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI362	2:0:2:0	3	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Learn the evolution of the World Wide Web and its relevance in today's world.
- Get a clear understanding of the technologies involved in developing a website.
- Learn web technology with a focus on creating interactive and responsive web pages.

Syllabus

Module - I

HTML 5 and CSS: Introduction to Hyper Text Markup Language, Key components of HTML document, HTML elements, Headers, Linking, Images, Unordered Lists, and Nested and ordered Lists. Tables, Divs and forms: HTML Tables and Formatting, HTML Forms, Internal Linking, Creating and Using Images, Maps, Div and span tags. Introduction CSS, CSS selector, positioning, layouts, debugging. **05 Hours**

Module - II

JavaScript: Browser and Document object, scripts and HTML Document, variables, expressions, Data type conversions, decisions and loops, control structure, windows Document object, forms and form handling elements, scripting, event handling. **06 Hours**

Module - III

jQuery: Using selectors with jQuery, Manipulating page elements with jQuery, jQuery event model, jQuery and Ajax, jQuery animation and advanced effects, jQuery plugins. **05 Hours**

Module - IV

Bootstrap: Bootstrap Scaffolding, Bootstrap CSS, Bootstrap Layout Components, Bootstrap Java Script Plugins, Using Bootstrap. **05 Hours**

Module - V

XML: What is XML? What are the differences between HTML and XML, what is the purpose of XML? AJAX: AJAX Introduction, AJAX XML Http, AJAX Request, AJAX Response, AJAX with Server side. **05 Hours**

Hands on

1. HTML and CSS -

a. Create a HTML page to display the following content <Ensure the format is same as shwn below>; Use HTML Tables UnorderList and OrderList (UL and OL):

1. Development Environment • Eclipse (SpringSource Tool Suite distribution) • Apache Tomcat/VMware® vFabric™ tc Server • Spring Insight • Testing tools	**2. Spring Overview** • Introduction to Spring configuration • Bean life cycle • Simplifying configuration • Integration testing with Spring
3. Getting Started with Spring Web MVC • Spring model-view-controller (MVC) overview • DispatcherServlet • Controller programming model overview • Spring MVC views • Simplifying configuration	**4. Spring MVC Configuration Options** • Spring MVC infrastructure Beans • URL mappings • Handler interceptors and handler adapters • Exception resolvers • Message source

b. Create a HTML page to display testimonials received from customers along with their picture and Name as shown below:

2. Java Script and jQuery

a. jQuery form validations:

Front-End: Develop below form using HTML to create new user:

Field Details:

1. Name <String, Length(16), Mandatory, Validations: Minimum Length: 3, Can accept special characters>
2. Email <String, Length(60), Mandatory, Validations: Should be a valid email id>
3. Password <String, Length(16), Mandatory, Validations: Minimum Length: 5, Can accept special characters>

- Create Buttons - "Create an account" and "Cancel"
- Form should be Scrollable
- For field validations, use jQuery
- Ensure all validations pertaining to Name, Email and Password are taken care. If the use enters incorrect values appropriate error message should be displayed and should allow the user to enter correct data

b. jQuery image slider:

In an HTML page, insert a minimum of 5 images; Ensure inserted images are scrollable.
Hint: To make images scrollable use jQuery image slider or use javascript.

Sample screen shot :

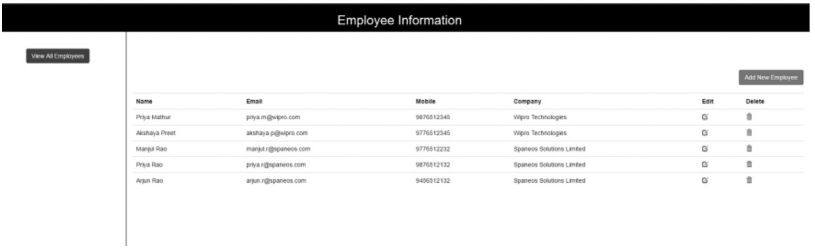
3. Develop below form using HTML to Search and Book Tickets:

Validations :

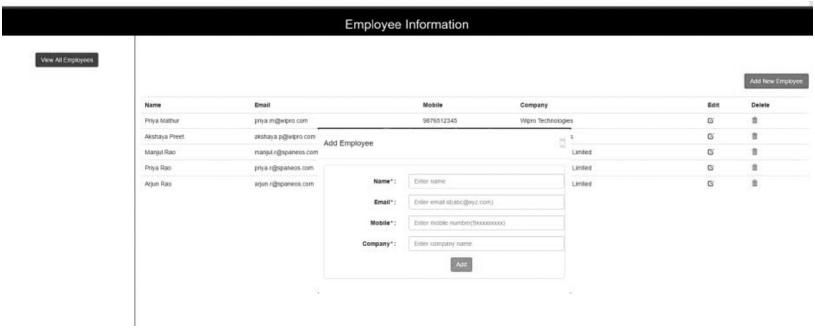
1. All fields are Mandatory except “Single Lady” Field
2. Onwards date must be less than Return date

4. Bootstrap, AJAX and jQuery:

a. Create a Bootstrap Page that helps maintain Employee Information in an organization.



b. When clicked on the “Add New Employee” button, load a dialog box as shown below



Validations:

1. All fields are Mandatory.
2. On successful submission of the form, the new employee details has to be appended as a last row in the table.
3. When clicked on the Edit icon, a similar pop-up as the "Add new Employee" form has to be displayed with the input fields populated with appropriate values. When the form is submitted in the Edit flow, ensure all the validations are in place. The name of the button in the Edit flow has to be "Update" instead of "Add".
4. When clicked on the Delete icon, a confirmation dialog box has to be displayed with a message "Are you sure, you want to delete this entry?" If the user clicks "Yes", the corresponding row has to be deleted from the table. If the user clicks "No" the table has to remain unaffected.

Course Outcomes:

On completion of this course, the students are able to :

- Develop web layouts with style sheets and web screens in a presentable form.
- Write interactive web pages through form validations and other methods. Use the same in UI development.
- Use the Java Script libraries to accelerate UI development.
- Design and develop responsive and mobile first web pages.
- Develop applications by using synchronous and asynchronous communication over web.

Text Book:

1. Jon Duckett: "Web Design with HTML, CSS, JavaScript and jQuery Set", Wiley, 1st Edition, 2014, ISBN 13: 978-1118907443.

Reference Books:

1. Jake Spurlock: "Bootstrap, Shroff", O'Reilly Media, United States of America, 1st Edition, 2013, ISBN: 978 -1 -4493-4391-0.
2. Bear Bibeault, Yehuda Katz and Aurelio De Rosa: "jQuery in Action", Dreamtech Press, New Delhi, India, 3rd Edition, 2015, ISBN: 978-1617292071.

E-Resources:

1. http://www.w3schools.com/
2. https://learn.jquery.com/
3. https://developer.mozilla.org/en-US/Learn/Getting_started_with_the_web/JavaScript_basics

Principles of Programming (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI363	2:0:2:0	3	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Understand and describe syntax and semantics of programming languages.
- Learn data, data types, and basic statements.
- Gain knowledge about call-return architecture and ways of implementing them.
- Understand object-orientation, concurrency, and event handling in programming languages.
- Develop programs in non-procedural programming paradigms.

Syllabus

Module - I

Syntax and Semantics: Evolution of programming languages, describing syntax, context-free grammars, attribute grammars, describing semantics, lexical analysis, parsing, recursive-decent, bottom-up parsing. **05 Hours**

Module - II

Data, Data Types, and Basic Statements: Names, variables, binding, type checking, scope, scope rules, lifetime and garbage collection, primitive data types, strings, array types, associative arrays, record types, union types, pointers and references, Arithmetic expressions, overloaded operators, type conversions, relational and boolean expressions, assignment statements, mixed-mode assignments, control structures, selection, iterations, branching, guarded statements. **06 Hours**

Module - III

Subprograms and Implementations: Subprograms, design issues, local referencing, parameter passing, overloaded methods, generic methods, design issues for functions, semantics of call and return, implementing simple subprograms, stack and dynamic local variables, nested subprograms, blocks, dynamic scoping. **05 Hours**

Module - IV

Object-Orientation, Concurrency and Event Handling: Object-orientation, design issues for OOP languages, implementation of object-oriented constructs,
concurrency,semaphores, monitors, message passing, threads, statement level concurrency, exception handling, even handling. **05 Hours**

Module - V

Functional and Logic Programming Languages: Introduction to lambda calculus, fundamentals of functional programming languages, Programming with Scheme, Programming with ML, Introduction to logic and logic programming, Programming with Prolog, multi-paradigm languages. **05 Hours**

List of Experiments:

1. Implement all major functions of string.h in single C program using switch case to select specific function from user choice (like strlen, strcat, strcpy, strcmp, strrev)
2. Write a program (WAP) in C to reverse a linked list iterative and recursive.
3. WAP in C to implement iterative Towers of Hanoi.
4. WAP in C++ to count the no.'s of object of a class with the help of static data member, function and constructor.
5. WAP in C++ and Java to declare a class Time with data members mm for minutes, ss for seconds and hh for Hours. Define a parameterize constructor to assign time to its objects. Add two time objects using member function and assign to third objects. Implement all possible cases of time.
6. WAP in C++ to define a class Complex to represents set of all complex numbers. Overload '+' operator to add two complex numbers using member function of the class and overload '*' operator to multiply two complex numbers using friend function of the class complex.

Course Outcomes:

On completion of this course, the students are able to :

- Describe syntax and semantics of programming languages.
- Explain data, data types, and basic statements of programming languages.
- Design and implement subprogram constructs.
- Apply object-oriented, concurrency, and event handling programming constructs.
- Develop programs in Scheme, ML, and Prolog.

Text Book:

1. Robert W. Sebesta: "Concepts of Programming Languages", 10th Edition, Addison Wesley, 2012, (Chapters 3,5-10,12-16) ISBN: 0-13-607347-6.

Reference Books:

1. R. Kent Dybvig: "The Scheme programming language", 4th Edition, MIT Press, 2009, ISBN 978-0-262-51298-5.

2. Richard A. O'Keefe: "The craft of Prolog", MIT Press, 2009, ISBN: 0 262 15039 5.
3. Michael L. Scott: "Programming Language Pragmatics", 3rd Edition, Morgan Kaufmann, 2009, ISBN-13: 978-0123745149, ISBN-10: 0123745144.

E- Resources:

1. https://www.google.co.in/?gfe_rd=crandamp
2. https://www.amazon.com/Programming-Language-Pragmatics
3. https://www.google.co.in/?gfe_rd=crandamp;ei=1r5PV7SWFtWQ2ASd3q3ADw#q=programming+language+pragmatics+morgan+kaufmann+pdf

Analog and Digital Electronics Laboratory

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISL37	1:0:2:0	2	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Learn skills to design and analyze electronic circuits and simulate the same using Multisim simulation package.
- Transform the simple logical equations.
- Perform simple jobs in wiring logical elements.

Syllabus

PART - A

1. a) Design and construct a suitable circuit and demonstrate the working of pos tive clipper, double-ended clipper and positive clamper using diodes.

   b) Demonstrate the working of the above circuits using a simulation package (Virtual lab).

2. a) Design and construct a suitable circuit and determine the frequency response, input impedance, output impedance, and bandwidth of a CE amplifier.

   a) Design and build the CE amplifier circuit using a simulation package and determine the voltage gain for two different values of supply voltage and for two different values of emitter resistance (Virtual lab).

3. a) Design and construct a Schmitt trigger using OP-AMP for given UTP and LTP values and demonstrate its working.

   a) Design and implement a Schmitt trigger using OP-AMP using a simulation package for two sets of UTP and LTP values and demonstrate its working (Virtual lab).

4. a) Design and construct a rectangular waveform generator (OP-AMP relaxation oscillator) for given frequency and demonstrate its working.

   a) Design and implement a rectangular waveform generator (OP-AMP relaxation oscillator) using a simulation package and demonstrate the change in frequency when all resistor values are doubled (Virtual lab).

5. Design and implement an astable multivibrator circuit using 555 timer for a given frequency and duty cycle.

PART – B

6. a) Given a 4-variable logic expression, simplify it using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC.

   b) Design and develop the Verilog / VHDL code for an 8:1 multiplexer. Simulate and verify its working (Virtual lab).

7. a) Realize a J-K Master / Slave Flip-Flop using NAND gates and verify its truth table.

   b) Design and develop the Verilog / VHDL code for D Flip-Flop with positive edge triggering. Simulate and verify its working (Virtual lab).

8. a) Design and implement a mod-n (n<8) synchronous up counter using J-K Flip-Flop ICs and demonstrate its working.

   a) Design and develop the Verilog / VHDL code for mod-8 up counter. Simulate and verify its working (Virtual lab).

9. a) Design and implement a ring counter using 4-bit shift register and demon strate its working.

   a) Design and develop the Verilog / VHDL code for switched tail counter. Simu late and verify its working (Virtual lab).

10. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n (n<=9) and demonstrate its working.

Note: In the examination students have to write and execute one experiment from a lot of the above 10 experiments.

Scheme of Practical Examination:

Conduction of two experiments	- 40 marks
Viva Voce	- 10 marks
Total	- 50 marks

Course Outcomes:

On completion of this course, the students are able to :

1. Design and analyze the electronic circuits and simulate the same using software tool.
2. Interpret basic binary math operations using the logic gates.
3. Develop skills on programming proficiency using the various logical elements to design practically motivated logical units.

Virtualization Foundations (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI38	1:0:2:0	2	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand the types of hypervisors and its features.
- Create a blank virtual machines using ESXi.
- Access VMware vCenter Server and the VMware ESXi hosts using vSphere Client.
- Create virtual networking and accessing storage.
- Convert a physical computer to Virtual Machine.
- Get a strong foundation for mastering computer virtualization.

Syllabus

Module - I

Introduction to Virtualization: Definition of Virtualization, Traditional versus Virtualized System, How Virtualization works, Need for Virtualization, Benefits of Virtualization, Definition of Hypervisor, Type-1 Hypervisor, Type-2 Hypervisor, Comparison of Type1 and Type2 Hypervisors, Types of Virtualization – Server Virtualization, Desktop Virtualization, Application Virtualization, Network Virtualization, Storage Virtualization. **06 Hours**

Module - II

VMware Player: Overview, Introduction to VMware Player, System Requirements, Installation of VMware Player, Enabling VT-x/AMD-v in BIOS, Creating a blank Virtual Machine, Installation of OpenSuse 64-bit, Introduction to Virtual Appliance, Demonstration of Virtual Appliance, Introduction to Unity mode, Demonstration of Unity mode. **05 Hours**

Module - III

ESXi: Introduction to ESXi, ESXi Architecture, System Requirements, Installing ESXi, Installing Windows 2008 R2, Installing vSphere Client, Creating a VM on ESXi using vSphere Client, Installing a Guest OS on ESXi. **05 Hours**

Module - IV

Networking and Storage: Introduction to Virtual Networking, Virtual Networking - Components and Concepts, Virtual Networking in ESXi, Introduction to Storage, How Virtual Machines Access Storage, Types of Physical Storage. **05 Hours**

Module - V

VMware Converter: Overview, Installation of VMware Converter, Converting a Physical Computer to a VM, Powering-on the converted VM. **05 Hours**

Course Outcomes:

On completion of this course, the students are able to :

- Explain what virtualization is, and get an in-depth understanding of how things work at the ring level.
- Learn how to install VMware Player and create virtual machines using VMware Player. In specific, the module will guide on setting up a virtual lab environment using VMware Player.
- Learn more about installing VMware ESXi, working with the vSphere Client, and managing a host using the vSphere Client.
- Cover the concepts of Virtual Networking and Storage with reference to ESXi.
- Learn how to use VMware Converter to convert physical computer into a virtual machine.

Reference Books:

1. Matthew Portnoy : "Virtualization Essentials", Wiley, 2012, ISBN: 978-1-118-17671-9.
2. Nelson Ruest, Danielle Ruest: "Virtualization, A Beginner's Guide", McGraw-Hill Education, ISBN-13: 978-0071614016.

Integrated Rural Development – Part 1

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISH39	0:2:0:0	1	CIE:50 SEE:50	2 Hours	HSS

Course Objectives:

This course will enable students to:

- Gain an awareness of the existing challenges in rural areas of India
- Develop the ability to communicate and interact with rural sections of our society
- Use and apply their academic knowledge to facilitate rural development and uplift via targeted initiatives and activities.

Syllabus

Module - I

Introduction: Introduction to the course and its objectives; overview of typical challenges faced in villages; importance of integrating villages in mainstream society; relevance of course to nation building; division of students into groups; allotment of villages to student groups; assignment of mentors to student groups. **03 Hours**

Module - II

Project Definition: Visit of student groups to respective villages with assigned mentors; interacting with villagers and ice-breaking activities; identifying possible project topics with the help of mentor and supervisor; student group discussion to finalize the project definition; review of project definition with mentor and supervisor. **06 Hours**

Module - III

Project Conceptualization and Planning: Creation of plan to realize the project; review of plan with mentor and supervisor; assigning action items to students within the group; planning for needed logistics and infrastructure. **06 Hours**

Module - IV

Project Realization: Execution of the project plan (for example by conducting workshops); aggregation of project deliverables like survey reports, collected data, interviews, and questionnaires; recording of impact of the project on the village; periodical review of the project execution status as well as the project deliverables (like aggregated data and survey reports) with mentor and supervisor. **10 Hours**

Module - V

Project Reporting: Creation of project report by the student groups detailing the motivation for the project, the approach, the work packages along with student assignments, the execution of the project, impact of the project, and lessons learned by the students during the project; creation of a slide-set to present the project report during the final exam; review by mentor and supervisor. **03 Hours**

Course Outcomes:

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

- Develop the ability to interact and communicate with different sections of society, thus improving their communication skills.
- Understand the existing problems and needs of a village, thus developing an awareness of the challenges facing rural India.
- Conceptualize, plan, and realize measures to address these problems, thus improving their practical problem-solving and leadership skills.
- Make an impact to rural section of society, thus building their self-confidence.

Text Books:

1. Bhagawan Sri Sathya Sai Baba: "Service to Village is Service to God", Sri Sathya Sai Publications.

Reference Books:

1. Bhagawan Sri Sathya Sai Baba: "Man Management: A Value-Based Management Perspective", Sri Sathya Sai Publications.
2. Lt. Gen. M.L.Chibber:"Sai Baba's Mahavakya on Leadership : Book for Youth, Parents and Teachers."

E-Resources:

1. http://rural.nic.in/netrural/rural/index.aspx
2. www.annapoorna.org.in

Engineering Mathematics-IV (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISM41	3:0:2:0	4	CIE:50 SEE:50	3 Hours	BS

Course Objectives:

This course will enable students to :

- Develop the application of mathematical skills in solving statistics and probability problems using computers.
- Use of Transforms in the engineering problems.
- Analyse the complex variables and functions
- Introduction of Statistical Software's.

Syllabus

Module - I

Random Variables: Discrete probability distribution, Continuous probability distribution, Expectation, Variance, Probability generating function, Binomial distribution, Poisson distribution, Exponential distribution and Normal distribution.

08 Hours

Module - II

Joint Probability: Joint probability distribution, discrete and independent random variables, expectation, covariance, correlation coefficient, probability vectors, stochastic matrices, fixed point matrices, regular stochastic matrices, Markov chains, higher transition probabilities, stationary distribution of regular Markov chains and absorbing states. **08 Hours**

Module - III

Complex variables: Functions of a complex variable, derivative of complex functions. **Analytic function**s: Cauchy's-Riemann equations in Cartesian and polar forms (No problems by using limits), Harmonic functions, construction of analytic functions by using Milne-Thomson method. Cauchy Theorem, Cauchy's integral formula-problems. **08 Hours**

Module - IV

Fourier Transforms: Infinite Fourier Transform, Fourier Sine and Cosine Transform and their inverse transforms-Problems.

Z-Transforms: Definition, Standard functions, statements of Linearity property, Damping and shifting rules-problems. Inverse Z-Transforms by partial fraction method. Difference equations– solutions by Z-transform. **08 Hours**

Module - V

Introduction to R, Basic Data types, vector operations, matrix construction, lists, data frames, Elementary statistics with R-Qualitative and Quantitative data, Numerical measures, probability distribution, interval estimation and simple linear regression.

08 Hours

List of R-Lab Experiments

Sl. No.	Name of the Experiment
1	Introduction to R Software and basic commands
2	Demonstration and operations of Vectors
3	Operations of Matrices
4	Demonstration of Lists
5	Demonstration of Data Frames
6	Qualitative Data Analysis
7	Quantitative Data Analysis
8	Numerical Measures of Data
9	Probability Distribution
10	Linear Regressions

Course Outcomes:

On completion of this course, the students are able to :

- Find the probability distribution of random experiments.
- Apply the stochastic process and Markov chain in prediction of future events.
- Evaluate the complex function differentiation and integration.
- Find Fourier and Z-transform of different function.
- Use the statistical software R for analyzing the data .

Text Books:

1. Dr. B.S. Grewal: "Higher Engineering Mathematics", Khanna publishers, New Delhi, 42nd Edition, 2012, ISBN: 9788174091956.
2. R.E. Walpole, R.H.Myers.R.S, L.Myers and K.Ye: "Probability and Statistics for Engineers and Scientists", Pearson Education, Delhi, 8 th Edition, ISBN 13: 9780131877115.
3. W.N.Venables, D.M.Smith : "An introduction to R".

Reference Books:

1. Erwin Kreyszig: "Advanced Engineering Mathematics", (Chapters 13,14,19,21,24,25), Wiley Pvt. Ltd., India, New Delhi, 9th Edition, 2011, ISBN 13: 9788126531356.
2. John Verzani: "Using R for introductory Statistics", Champan and Hall/ CRC, New York, Washington D.C., ISBN: 978-1-59327-384-2.
3. Sheldom M Ross: "Probability models for Computer Science", Academic Press, 2009, ISBN: 9780124079489.
4. Murray R Spiegel, John Schiler and Alu Srinivasan: "Probability and Statistics", Schaum's Outline series, 2nd Edition, ISBN: 9780071795579.

E-Resources:

1. http://www.zums.ac.ir/ebooks/mathematics/essential-engineering-mathematic.
2. https://archive.org/details/AdvancedEngineeringMathematics10thEdition
3. https://www.r-project.org/
4. www.r-tutor.com

Formal Languages and Automata Theory

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16IST42	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course objectives:

This course will enable students to :

- Understand the basic concepts of finite automata.
- Learn the regular expressions, languages and their applications.
- Describe the context free grammars, languages and their applications.
- Acquire knowledge about push down automata, languages and its types.
- Introduce Turing machines

Syllabus

Module - I

Introduction to Finite Automata: Introduction to Finite Automata, The central concepts of Automata theory, Deterministic finite automata, Nondeterministic finite automata. **08 Hours**

Module - II

Finite Automata, Regular Expressions: An application of finite automata, Finite automata with Epsilon-transitions, Regular expressions, Finite Automata and Regular Expressions, Applications of Regular Expressions. **08 Hours**

Module - III

Regular Languages, Properties of Regular Languages: Regular languages, Proving anguages not to be regular languages, Properties of regular languages, Equivalence and minimization of automata. **07 Hours**

Module - IV

Context-Free Grammars And Languages: Context –free grammars, Parse trees, Applications, Ambiguity in grammars and Languages.

Properties of Context-Free Languages: Normal forms for CFGs, Pumping lemma for CFGs , Properties of CFLs. **09 Hours**

Module - V

Pushdown Automata: Pushdown automata, the languages of a PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automata.

Introduction To Turing Machine: Problems that computer cannot solve, the Turing machine. **08 Hours**

Course Outcomes:

On completion of this course, the students are able to :

- Design the different types of automata for given regular expression and regular languages.
- Write the grammar for the given regular expressions.
- Draw the parse tree for the given context free grammars.
- Design the PDA for the given languages.
- Convert the given PDA to its equivalent CFG and vice versa.

Text Book:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman: "Introduction to Automata Theory, Languages and Computation", (Chapters 1-8), 3rd Edition, Pearson New International Edition, 2014, ISBN-13: 978-0321455369.

Reference Books:

1. Peter Linz: "An Introduction to Formal Languages And Automata", 4th Edition, 2006, ISBN-13: 978-0763737986, ISBN-10: 0763737984.
2. K.L.P. Mishra: "Theory of Computer Science, Automata Languages and Computation", 3rd Edition, PHI Learning, 2009, ISBN 10: 8120329686, ISBN 13: 9788120329683.

E-Resources:

1. http://www.ebook777.com/theory-finite-automata-introduction-formal-languages
2. http://www.techmela.ucoz.com/_ld/0/22_Introduction_to.pdf

Design and Analysis of Algorithms

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16IST43	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- To learn good principles of algorithm design.
- To analyze algorithm and estimate their performance.
- To distinguish the lower and upper bounds of various problems and their importance in deciding the optimality of an algorithm.

Syllabus

Module - I

Introduction: Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Review of Asymptotic Notations, Mathematical Analysis of Non-Recursive and Recursive Algorithms, Brute Force Approaches: Introduction, Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching, Closest-Pair and Convex-Hull Problems by Brute Force, Exhaustive Search. **08 Hours**

Module - II

Divide and conquer: Divide and Conquer: General Method, Binary Search, Merge Sort, Quick Sort and its performance, Binary tree traversal methods and related properties. **08 Hours**

Module - III

Greedy method: The General Method, Minimum-Cost Spanning Trees: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's algorithm, Huffman trees. **08 Hours**

Module - IV

Dynamic programming: The General Method, Warshall's Algorithm, Floyd's Algorithm for the All-Pairs Shortest Paths Problem, Single-Source Shortest Paths: General Weights, 0/1 Knapsack, The Traveling Salesperson problem. **08 Hours**

Module - V

Decrease-and-conquer approaches, space-time tradeoffs: Decrease-and-Conquer Approaches: Introduction, Insertion Sort, Depth First Search and Breadth First Search, Topological Sorting Space-Time Tradeoffs: Introduction, Sorting by Counting, Input Enhancement in String Matching, Hashing, B-Trees. **08 Hours**

Course Outcomes:

On completion of this course, the students are able to :

- Analyze the algorithm by estimating best, average and worst case time complexities.
- Apply divide and conquer techniques.
- Apply greedy method for getting optimal solution.
- Illustrate dynamic programming paradigm.
- Apply decrease and conquer techniques and analyze space and time tradeoffs.

Text Book:

1. Anany Levitin: "Introduction to the Design and Analysis of Algorithms", 2nd Edition, Pearson Education, 2011, ISBN-10: 0132316811, (Chapters 1,3,4.1 to 4.4, 5.1 to 5.3,7,8,9).

Reference Book:

1. Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein: "Introduction to Algorithms", 3rd Edition, PHI, 2010. ISBN-13: 9780262033848.

E-Resources:

1. http://www.pearsoned.co.in/prc/book/anany-levitin-introduction-design-analy-sis-algorithms-2e-2/9788131718377
2. https://www.scribd.com/doc/59300888/Introduction-to-the-Design-Analysis-of-Algorithms

UNIX and Shell Programming (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI44	3:0:2:0	4	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand the UNIX architecture and file system in UNIX and various commands.
- Understand the basic file attributes, vi editor, and process.
- Get exposed to shell programming.

Syllabus

Module - I

Introduction: The UNIX operating system, the UNIX attributes, features of UNIX, locating commands, internal and external commands, flexibility and command usage, man command.

The file system: The file, what is a filename, parent child relationship, the home variable, pwd, cd, mkdir, commands, absolute and relative pathnames ls, the Unix file system.

Handling ordinary files: cat, cp, rm, od, wc, comp comm, diff. **08 Hours**

Module - II

Basic file attributes: ls –l, the –d option, file ownership, file permission, chmod: changing file permission, directory permission, changing file ownership.

The vi Editor: vi basics, input mode, ex mode, navigation, editing text, repeating the last command, searching for pattern, substitution.

The shell: The shell interpreter cycle, shell offerings, pattern matching, escaping and quoting redirection, /dev/tty the 2 special files, pipes, tee command, substitution, shell variables. **08 Hours**

Module - III

The Process: Process basics , ps, system processes (-p or –c) mechanism of process creation, internal and external commands, running jobs in back ground, killing processes with signal, job control.

Customizing the environment: Environment variable.

More file attributes: File system and inodes, hard links, symbolic link, ln, the directory,

umask, modification and access time, time. Simple filter: pr, head, cut, paste, sort, uniq, tr. **08 Hours**

Module - IV

Filters using regular expression: grep, basic regular expression, extended regular expression and egrep, sed –line addressing, context addressing and substitution. **Essential shell programming:** Shell scripts, read, command line argument, exit and exit status of command, logical operators, the case conditions, expr, compilation and string handling. **08 Hours**

Module - V

Essential shell programming and awk: while, for, set and shift, the here document, trap. awk an advanced filter**:** simple awk filtering, splitting line into fields, comparison operators, the BEGIN and END section, built in variable, arrays, function, the if statement, for and while. **08 Hours**

Laboratory Experiments

Execute the following programs:

1. Non-recursive shell script that accepts any number of arguments and prints them in the Reverse order, (For example, if the script is named rargs, then executing rargs A B C should produce C B A on the standard output).
2. Shell script that accepts two file names as arguments, checks if the permissions for these files are identical and if the permissions are identical, outputs the common permissions, otherwise outputs each file name followed by its permissions.
3. Shell script that accepts path names and creates all the components in that path names as directories. For example, if the script name is mpe, then the command mpe A/B/C/D should create directories A, A/B, A/B/C, and A/B/C/D.
4. Shell script that accepts valid login names as arguments and prints their corresponding home directories. If no arguments are specified, print the suitable message.
5. Shell script that takes a valid directory names as an argument and recursively descends all sub directories, find the maximum length of any file in that hierarchy and prints this maximum value to standard output.
6. Shell script that accepts file names specified as arguments and creates a shell script that contains this file as well as the code to recreate these files. Thus if the script generated by your script is executed, it would recreate the original files (This is same as the “bundle” script described by Brain W. Kernighan and Rob Pike in “ The Unix Programming Environment”, Prentice – Hall India).
7. Shell programming of execute multiple shell commands using suitable like ls –l , ps -f, date , cal etc.
8. Shell script the reports in descending order of their size names, names and sizes of files whose size exceeds 40 bytes, in a specific directory (supplied as an argument) total numbers of a search files is also displayed.

9. Shell script to implement terminal locking. It should prompt user for password. After accepting password entered by user, it must prompt again for password confirmation (to retype the password). if a match occurs, it must lock terminal and prompt for the password. If proper password is entered, the terminal must be unlocked.

Course Outcomes:

On completion of this course, the students are able to :

- Analyze the Architecture of UNIX operating system and know the basic UNIX commands.
- Execute commands required for manipulation of file and its attributes.
- Write shell script to create process, control process and implement simple filters.
- Analyze the various UNIX commands, advanced filters and regular expressions for file processing.
- Implement shell programs, awk and sed.

Text Book:

1. Sumitabha Das: "UNIX – Concepts and Applications", (Chapters 1.2,2,4,6-14), 4th Edition, Tata McGraw Hill, 15th Reprint 2011, ISBN 13: 9780070635463.

Reference Books:

1. Behrouz A. Forouzan and Richard F. Gilberg: "UNIX and Shell Programming", Cengage Learning, 2005, ISBN: 0 534-95159-7.
2. M.G. Venkateshmurthy: "UNIX and Shell Programming", Pearson Education, 2005, ISBN: 9788177587456.

E-Resources:

1. http://people.ischool.berkeley.edu/kevin/unix-tutorial/toc.html
2. http://www.tutorialspoint.com/unix/unix_tutorial.pdf
3. https://books4cse.files.wordpress.com/2014/01/sumitabahdas.pdf

Introduction to Microprocessors (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI451	3:0:2:0	4	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Architecture of microprocessor 8086.
- Mathematical computations and programming using 8086.
- 8086 Memory interfacing and understanding of 8255.
- Different types of interrupts and their applications in 8086.

Syllabus

Module - I

Introduction to Microprocessors: Internal Microprocessor Architecture (8086 to Pentium), Flag register of 8086, Real mode memory addressing, Pin outs and Pin functions of 8086, Bus buffering and latching, Bus timing, Ready and Wait state, **Addressing modes:** Data, Program memory, Stack memory. **08 Hours**

Module - II

Instruction set of 8086: Data move, Arithmetic and Logic, Program control. Assembler directives, Assembly language programming, Programs using BIOS and DOS interrupts, assembly language programming with C/C++ for 16 bit applications. **08 Hours**

Module - III

Memory interfacing: Address decoding, Static memory interfacing with 8086. Introduction to dynamic memory interfacing, Introduction to I/O interface, I/O port address decoding (8 bit and 16 bit). Simple programs related to I/O interface. **08 Hours**

Module - IV

PPI: Study of 8255: Control word, different modes of operation. Study of 8279. Interfacing programs with: Stepper motor, Keyboard, Display, Timer and ADC/DAC interfaces. **08 Hours**

Module - V

Basic Interrupt processing, Hardware interrupts, expanding the interrupt structure, interrupt examples. Basic DMA operations, study of 8237 DMA controller, Introduction to MMX technology. **08 Hours**

List of Experiments

PART A:

1. Search a key element in a list of 'n' 16-bit numbers using the Binary search algorithm.
2. Write two ALP modules stored in two different files; one module is to read a character from the keyboard and the other one is to display a character. Use the above two modules to read a string of characters from the keyboard terminated by the carriage return and print the string on the display in the next line.
3. Sort a given set of 'n' numbers in ascending order using the Bubble Sort algorithm.

4. Read an alphanumeric character and display its equivalent ASCII code at the center of the screen.
5. Reverse a given string and check whether it is a palindrome or not.

PART B:

1. Read the status of eight input bits from the Logic Controller Interface and display 'FF' if it is the parity of the input read is even; otherwise display '00'.
2. Display messages FIRE and HELP alternately with flickering effects on a 7-segment display interface for a suitable period of time. Ensure a flashing rate that makes it easy to read both the messages (Examiner does not specify these delay values nor is it necessary for the student to compute these values).
3. Scan an 8 x 3 keypad for key closure and to store the code of the key pressed in a memory location or display on screen. Also display row and column numbers of the key pressed.
4. Drive a Stepper Motor interface to rotate the motor in specified direction (clockwise or counter-clockwise) by N steps (Direction and N are specified by the examiner). Introduce suitable delay between successive steps. (Any arbitrary value for the delay may be assumed by the student).
5. Generate the Sine Wave using DAC interface (The output of the DAC is to be displayed on the CRO).

Course Outcomes:

On completion of this course, the students are able to :

- Explain the Intel 8086 architecture, pin functions and bus timing diagrams.
- Develop 8086 assembly language programs for different applications.
- Design memory and I/O interfacing circuits with the help of PPI to the 8086 processor.
- Develop 8086 assembly language programs for different interfacing boards.
- Formulate interrupt programs with 8086 hardware and software interrupt methods.

Text Book:

1. Barry B Brey: "The Intel Microprocessors-Architecture, Programming and Interfacing", 8th Edition, Pearson Education, Kindle Edition, 2013, ISBN-13: 978-0135026458. (Only listed topics from Chapters 2, 3, 7-14).

Reference Books:

1. A.K Ray, K.M.Bhurchandi : "Advanced Microprocessors and Peripherals", 2nd Edition, TMH, 2006, ISBN 13: 9780070140622.
2. Uffen Beck: "8086: Architecture and Interfacing", 2nd Edition, John Wiley, 2005.

E-Resources:

1. https://en.wikipedia.org/wiki/Intel_8086
2. http://elearning.vtu.ac.in/elcmys/13/ENotes/8086/unit%201.pdf
3. http://faculty.ksu.edu.sa/djemal/EE353/EE353Chap2-1.0.pdf

Object Oriented Programming with C++ (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI452	3:0:2:0	4	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Features of object oriented Programming concepts.
- Inline functions, default arguments, classes and objects.
- Constructor, Types of constructor and destructor and their order of execution.
- Operator overloading and its necessity.
- The virtual function, polymorphism, exception handling.

Syllabus

Module - I

Introduction: Origin of C++, features of OOP, Sample C++ program, Different data types, operators, expressions, implicit conversion, Type cast operator and statements, arrays and strings, pointers and user defined types, reference variable, memory management operator, name space, control structure, Function, default argument, in line functions, function overloading, recursive functions. **08 Hours**

Module - II

Classes and Objects: Classes, structures and classes are related, Friend functions, in line functions, Constructors, Different types of constructor, Destructors, Static data members, when constructor and destructors are executed, scope resolution operator. Nested classes, local classes, passing objects to functions, returning objects, this pointer. **08 Hours**

Module - III

Inheritance: Base Class, Inheritance, Types of inheritance and protected members, protected base class inheritance, inheriting multiple base classes, Constructors, Destructors and inheritance, Passing parameters to base class constructors, Granting access, Virtual base classes. **08 Hours**

Module - IV

Virtual functions, Polymorphism and Operator overloading: Operator overloading basics, creating a member operator function, Operator overloading using friend functions such as +, - , pre-increment, post-increment, etc., overloading <<, >>. Virtual function, calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, Pure virtual functions, Abstract classes, Using virtual functions, Early and late binding. **09 Hours**

Module - V

Generic function ,Exception handling C++ File I/O: Generic function, a function with two generic types, Generic sort. Exception handling fundamentals, catching class

types, using multiple catch, catching all exception.<stream>, and the file classes, opening and closing file, reading and writing text files, put(), get(), read(), write(), geline(), eof(), seekg(), seekp(), tellp(), tellg(). **07 Hours**

List of Experiments

Write and execute a program in C++ based on the following requirements:

1. An EMPLOYEE class is to contain the following data members and member functions: Data members: Employee Number (an integer), Employee Name (a string of characters), basic_ Salary (an integer), All Allowances (an integer), IT (an integer), Net Salary (an integer). Member functions: to read the data of an employee, to calculate Net_Salary and to print the values of all the data members. (All_Allowances = 123% of Basic; Income Tax (IT) = 30% of the gross salary (= basic_Salary- All_Allowance); Net_Salary = Basic_Salary + All_Allowances – IT.
2. Write and execute C++ program to create class called STUDENT with data members USN, NAME, AGE. Using inheritance creates the class UGSTUDENT and PGSTUDENT having fields as semester, fees and stipend, enter the data for at least 5 students. Find the semester wise average age of all UG and PG students separately.
3. Write and execute a program in C++ to create a class called COMPLEX and implement the following overloading function ADD that returns complex number.
   - ADD (a, c1)-where 'a' is integer and c1 complex number.
   - ADD (c1, c2)-where 'c1' and 'c2' are complex numbers.
4. Write and execute a program in C++ to create a class called STACK using an array of integers and to implement the following operations by overloading the operators + and - :
   - s1=s1 + element; where s1 is an object of the class STACK and element is an integer to be pushed on to top of the stack.
   - s1=s1- ; where s1 is an object of the class STACK and – operator pops off the top element.
5. Write and execute a program in C++ to create a class called DATE with methods to accept two valid dates in the form dd/mm/yy and to implement the following operations by overloading the operators + and -. After every operation the results are to be displayed by overloading the operator <<.
   - no_of_days = d1 – d2; where d1 and d2 are DATE objects, d1 >=d2 and no_of_ days is an integer.
   - d2 = d1 + no_of_days; where d1 is a DATE object and no_of_days is an integer.

6. Write and execute a program in C++ to create a class called STRING and implement the following operations. Display the results after every operation by overloading the operator <<.
   - STRING s1 = "VTU"
   - STRING s2 = "BELGAUM"
   - STIRNG s3 = s1 + s2; (Use copy constructor)
7. Write a C++ program to create base class called Geometric figure which contains a virtual function called Area (). Two new classes namely Triangle and Rectangle, Area function will calculate area of given geometric figure
8. Write and execute a program in C++ to create a class called MyException which prompts for user for positive number, if negative number is entered the class MyException should display error.

Note: In the examination each student executes one program from a lot of all the 8 Questions.

Course Outcomes:

On completion of this course, the students are able to :

- Apply the concepts of object oriented programming.
- Implement the concepts of classes and objects.
- Apply the concepts of inheritance to solve complex problems.
- Implement mechanism of virtual function and polymorphism.
- Develop generic function to perform different operations on different data types and implement exception handling.

Text Book:

1. Herbert Schildt: "The Complete Reference C++", (Chapters 11-21), 4th Edition, Tata McGraw Hill, 2003, ISBN 13: 9780070532465.

Reference Books:

1. Stanley B.Lippmann, Josee Lajore: "C++ Primer", 4th Edition, Pearson Education, 2005, ISBN-10: 0-321-71411-3.
2. Paul J Deitel, Harvey M Deitel: "C++ for Programmers", Pearson Education, 2009, ISBN-10: 0137059663.

E-Resources:

1. http://www.tutorialspoint.com/cplusplus/cpp_tutorial.pdf
2. http://www.ddegjust.ac.in/studymaterial/mca-3/ms-17.pdf.

Introduction to Programming using Python (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI453	3:0:2:0	4	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Understand programming concepts, and various programming paradigms.
- Get a clear understanding of Object Oriented Programming.
- Learn Python with a focus on regular expressions, exception handling, file handling, creating modules, interacting with database.

Syllabus

Module-I

Introduction and overview Introduction, What is Python, Origin, Comparison, Comments, Operators, Variables and Assignment, Numbers, Strings, Lists and Tuples, Dictionaries, if Statement, while Loop, for Loop and the range() Built-in Function, Files and the open() Built-in Function, Errors and Exceptions, Functions, Classes, Modules. Syntax and Style: Statements and Syntax, Variable Assignment, Identifiers, Basic Style Guidelines, Memory Management, Python Application Examples.

08 Hours

Module-II

Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types. Numbers and Strings Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions. Sequences: Strings, Lists, and Tuples, Sequences, Strings, Strings and Operators, String-only Operators, Built-in Functions, String Built-in Methods, Special Features of Strings.

08 Hours

Module-III

Lists Operators, Built-in Functions, List Type Built-in Methods, Special Features of Lists, Tuples, Tuple Operators and Built-in Functions, Special Features of Tuples. Conditionals and Loops if statement, else Statement, else if Statement, while Statement, for Statement, break Statement, continue Statement, pass Statement, else Statement. **08 Hours**

Module-IV

Files and Input/output File Objects, File Built-in Function, File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules Exception handling: The dir Function, Errors, Runtime Errors, The Exception Model, Exception Hierarchy, Handling Multiple Exceptions, raise, assert. **08 Hours**

Module-V

Regular Expressions Introduction/Motivation, Special Symbols and Characters for REs, REs and Python. Programming Exercise: Check for data error in CSV files: Numeric Check, Alphanumeric Check, Email Check, Date Check, Database Interactions: Database Connection, creating database tables, insert data into table, reading, updating data. **08 Hours**

List of Experiments:

1. Create a new program called hello world.py. You will use this file to write your very first 'Hello world!' program.
2. Write a program using print that, when run, prints out a tic-tac-toe board.
3. Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3,1/4... 1/10.
4. Write a program using a for loop that calculates exponentials. Your program should ask the user for a base base and an exponent exp, and calculate baseexp.
5. Write a method fact that takes a number from the user and prints its factorial.
6. Write a function roots that computes the roots of a quadratic equation. Check for complex roots and print an error message saying that the roots are complex.

Course Outcomes:

On completion of this course, the students are able to :

- Apply the concepts of Object Oriented principles used in Python.
- Apply Types, Type Operators and Built-in functions and use the same in developing specific programs.
- Apply the usage of built-in libraries, creation of customized libraries and efficient ways to store and retrieve data.
- Use file handling and exception handling mechanisms and apply the same in solving specific problems.
- Apply techniques using regular expressions and apply the same in solving specific problems.

Text Book:

1. Mark Lutz: "Learning Python", 5th Edition, O'REILLY, 2013, ISBN: 9781449355739

Reference Books:

1. Barry, Paul: "Head First Python", 2nd Edition, O'REILLY, 2010, ISBN: 978-1-4493-8267-4.
2. David M. Beazley: "Python Essential Reference", 4th Edition, Developer's Library,2010, ISBN: 0672329786.

E-Resources:

1. http://www.tutorialspoint.com/python/
2. https://www.codementor.io/learn-python-online
3. https://www.youtube.com/playlist?list=PL9FAE4422FA13FDE4

Cyber Security

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16IST461	3:0:0:0	3	CIE:50 SEE:50	3 Hours	EE

Course Objectives:

This course enable students to :

- Understand the area of cyber crime and forensics.
- Understand the motive and causes for cyber crime, detection and handling.
- Study the areas affected by cyber crime and investigation.
- Understand the tools used in cyber forensic.
- Know Legal Perspectives in cyber security.

Syllabus

Module - I

Introduction to Cyber crime: Cyber crime: Definition and Origins of the Word, Cyber crime and Information Security, Who are Cyber criminals?, Classifications of Cyber crimes, Cyber crime: The Legal Perspectives, Cyber crimes: An Indian Perspective, Cyber crime and the Indian ITA 2000, A Global Perspective on Cyber crimes, Cyber crime Era: Survival Mantra for the Netizens.

Cyber offenses: How Criminals Plan Them: How Criminals Plan the Attacks, Social Engineering, Cyber stalking, Cyber cafe and Cyber crimes, Bot nets: The Fuel for Cyber crime, Attack Vector, Cloud Computing. **08 Hours**

Module - II

Cyber crime: Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/CellPhones, Mobile Devices: Security Implications for organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era, Laptops. **08 Hours**

Module - III

Tools and Methods Used in Cyber crime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spy wares, Virus and Worms, Trojan Horses and Back doors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks.

Phishing and Identity Theft: Introduction, Phishing, Identity Theft (ID Theft).

08 Hours

Module - IV

Understanding Computer Forensics: Introduction, Historical Background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber

forensics and Digital Evidence, Forensics Analysis of E-Mail, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation, Setting up a Computer Forensics Laboratory: Understanding the Requirements, Computer Forensics and Steganography, Relevance of the OSI 7 Layer Model to Computer Forensics, Forensics and Social Networking Sites: The Security/Privacy Threats, Computer Forensics from Compliance Perspective, Challenges in Computer Forensics, Special Tools and Techniques, Forensics Auditing, Anti forensics. **08 Hours**

Module - V

Cyber crimes and Cyber security: The Legal Perspectives, Introduction, Cyber crime and the Legal Landscape around the World, Why Do We Need Cyber laws: The Indian Context, The Indian IT Act, Challenges to Indian Law and Cyber crime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Amendments to the Indian IT Act, Cyber crime and Punishment, Cyber law, Technology and Students: Indian Scenario.

08 Hours

Course Outcomes:

On completion of this course, the students are able to :

- Acquire knowledge about the cyber security cyber crime and cyber offenses.
- Explain cyber crime on various mobile and wireless devices.
- Use of tools and methods in cyber crime and security.
- Interpret computer forensics.
- Understand legal issues in cyber crime.

Text Book:

1. Sunit Belapure and Nina Godbole: "Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives", (Chapter 1-7), Wiley India Pvt. Ltd., New Delhi, India, 2011, ISBN: 8126521791.

Reference Book:

1. Dr. Surya Prakash Tripathi, Ritendra Goyal, Praveen Kumar Shukla, KLSI: "Introduction to information security and cyber laws", Dreamtech Press, New Delhi, India, 2014, ISBN 13: 789351194736.

E-Resources:

1. http://www.civilserviceindia.com/subject/General-Studies/notes/basics-of-cyber security.html
2. http://uttaminstitute.ac.in/CYBER.pdf
3. http://www.vssut.ac.in/lecture_notes/lecture1423183198.pdf
4. http://www.tutorialspoint.com/information_security_cyber_law/introduction.html

Renewable Energy Resources

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16IST462	3:0:0:0	3	CIE:50 SEE:50	3 Hours	EE

Course Objectives:

This course will enable students to :

- Provide detailed information of the present energy scenario and the available Renewable Energy Resources.
- Get detailed insight knowledge in basics of solar radiation geometry and various measurement techniques.
- Understand the solar energy through solar thermal devices, PV conversion and their performance analysis.
- Gain the conceptual knowledge about the various energy conversion methods such as Wind, Tidal, OTEC and Geothermal.
- Give introduction to Energy from Biomass, Hydrogen energy and their impact on environment and sustain ability.

Syllabus

Module - I

Introduction: Energy source, India's production and reserves of commercial energy sources, need for non-conventional energy sources.

Solar Radiation : Extra-Terrestrial radiation, spectral distribution of extraterrestrial radiation, solar constant, solar radiation at the earth's surface, beam, diffuse and global radiation, solar radiation data.

Measurement of Solar Radiation: Pyrometer, shading ring pyrheliometer, sunshine recorder, schematic diagrams and principle of working. **08 Hours**

Module - II

Solar Radiation Geometry: Flux on a plane surface, latitude, declination angle, surface azimuth angle, hour angle, zenith angle, solar altitude angle expression for the angle between the incident beam and the normal to a plane surface (No derivation), local apparent time. Apparent motion of sun, day length, numerical examples.

Radiation Flux on a Tilted Surface: Beam, diffuse and reflected radiation, expression for flux on a tilted surface (no derivations), numerical examples.

Solar Thermal Conversion: Collection and storage, thermal collection devices, liquid flat plate collectors, solar air heaters concentrating collectors (cylindrical, parabolic, paraboloid) (Quantitative analysis). **08 Hours**

Module - III

Performance Analysis of Liquid Flat Plate Collectors: General description, collector geometry, selective surface (qualitative discussion) basic energy-balance equation, stagnation temperature, transmissivity of the cover system, transmissivity – absorptivity product , numerical examples. The overall loss coefficient, correlation for the top loss coefficient, bottom and side loss coefficient, problems (all correlations to be provided). Temperature distribution between the collector tubes, collector heat removal factor, collector efficiency factor and collector flow factor, mean plate temperature, instantaneous efficiency (all expressions to be provided). Effect of various parameters on the collector performance; collector orientation, selective surface, fluid inlet temperature, number covers, dust. **08 Hours**

Module - IV

Photovoltaic Conversion: Description, principle of working and characteristics, applications.

Wind Energy: Properties of wind, availability of wind energy in India, wind velocity and power from wind, major problems associated with wind power, wind machines; Types of wind machines and their characteristics, horizontal and vertical axis wind mills.

Tidal Power: Tides and waves as energy suppliers and their mechanics, fundamental characteristics of tidal power, harnessing tidal energy, limitations. **Ocean Thermal Energy Conversion:** Principle of working, Rankine cycle.

Geothermal Energy Conversion: Principle of working, types of geothermal station with schematic diagram. **08 Hours**

Module - V

Energy from Bio Mass: Photosynthesis, photosynthetic oxygen production, energy plantation, bio gas production from organic wastes by anaerobic fermentation, description of bio-gas plants, transportation of bio-gas, problems involved with bio-gas production, application of bio-gas, application of bio-gas in engines, advantages.

Hydrogen Energy: Properties of Hydrogen with respected to its utilization as a renewable form of energy, sources of hydrogen, production of hydrogen, electrolysis of water, thermal decomposition of water, thermo chemical production bio-chemical production. **08 Hours**

Course Outcomes:

On completion of this course, the students are able to :

- Explain the present energy scenario and the available Renewable Energy Resources.
- Describe the basics of solar radiation geometry and various measurement techniques.

- Analyze the knowledge gained in tapping the solar energy through solar thermal devices, pv conversion and their performance analysis.
- Demonstrate the various energy conversion methods such as wind, Tidal, OTEC and Geothermal.
- Apply knowledge of Biomass and Hydrogen energy and their impact on environment and sustainability.

Text Books:

1. G D Rai: "Non-Conventional Energy Sources", (Chapters 1-3,6-9,11), Khanna Publishers, 5th Edition, 2011, ISBN-13: 9788174090737.
2. John Twidell and Tony Weir: "Renewable Energy Resources", (Chapters 2,5-7,9-14), Routledge Publisher, 3rd Edition, 2015, ISBN-13: 978041558437.
3. N K Bansal: "Non-Conventional Energy Resources", (Chapters-1-3,9,10,12,13), Vikas Publishing, 2014, 1st Edition, ISBN-13: 978935978577.

Reference Books:

1. B H Khan: "Non-Conventional Energy Resources", (Chapters 4-10), Tata McGraw-Hill Pub, 2nd edition, 2006, ISBN-13: 9780070142763.
2. S P Sukhatme, J K Nayak: "Solar Energy", (Chapters 3,4), Tata McGraw-Hill Pub., 3rd Edition, 2008, ISBN-13: 9780070260641.

E-Resources:

1. http://www.e-booksdirectory.com/details.php?ebook=9353.
2. https://books.google.co.in/books/about/NON_CONV_ENERGY_RES.html.
3. http://www.abebooks.com/book-search/author/s-sukhatme-j-nayak/.

Smart Materials

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16IST463	3:0:0:0	3	CIE:50 SEE:50	3 Hours	EE

Course Objectives:

This course will enable students to :

- Understand the characteristics of composites and smart materials in the product design process.
- Know the types of sensing and actuation devices.
- Gain the knowledge of optics and electromagnetic technology.
- Study the importance of different control systems.
- Realize and understand the principles of vibration and modal analysis.

Syllabus

Module - I

Introduction: Characteristics of composites and ceramics materials, Dynamics and controls, concepts, Electro-magnetic materials and shape memory alloys-processing and characteristics.

Control Design: Design of shape memory alloys, Types of MR fluids, Characteristics and application, principles of MR fluid value designs, Magnetic circuit design, MR Dampers, Design issues. **08 Hours**

Module - II

Sensing And Actuation: Principles of electromagnetic, acoustics, chemical and mechanical sensing and actuation, Types of sensors and their applications, their compatibility writer, conventional and advanced materials, signal processing, principles and characterization. **08 Hours**

Module - III

Structures: Principles of drag and turbulence control through smart skins, applications in environment such as aerospace and transportation vehicles, manufacturing, repair and maintainability aspects.

Optics And Electromagnetic: Principles of optical fiber technology, characteristics of active and adaptive optical system and components, design and manufacturing principles. **08 Hours**

Module - IV

Controls: Principles of structural acoustic control, distributed, analog and digital feedback controls, Dimensional implications for structural control. **08 Hours**

Module - V

Principles of Vibration and Modal Analysis: PZT Actuators, MEMS, Magnetic shape Memory Alloys, characteristics and Applications.

Information Processing: Neural Network, Data Processing, Data Visualization and Reliability – Principles and Application domains. **08 Hours**

Course Outcomes:

On completion of this course, the students are able to :

- Explain the characteristics of composites and smart materials in the product design process.
- Identify various types of sensing and actuation devices.
- Analyze the optics and design structures using smart materials.
- Demonstrate the working principles of different control systems.
- Describe the principles of vibration and modal analysis.

Text Books:

1. A V Srinivasan, D Michael Mcfarland: "Smart Structures: Analysis and Design", (Chapters 2-5,7,8), Cambridge University Press, 1st Edition, 2001, ISBN-13: 9780521659772.
2. M V Gandhi, B S Thomson: "Smart Materials and Structures", (Chapters 13-75), Chapman and Hall Pub., 1st Edition, 1992, ISBN-13: 9780412370106.

Reference Books:

1. Eric Udd: "Fiber Optic Sensors:An introduction for Engineers and Scientists", (Chapters 1-16), John Wiley and Sons Pub., 2nd Edition, 2011, ISBN-13: 9780470126844.
2. G P Gibss: "Adaptive Structures", John Wiles and Sons, New York, 1998.
3. Banks HT, RC Smith, Y Wang, Massow S A: "Smart Materials and Structures", Paris, 1996.

Design and Analysis of Algorithms Laboratory

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISL47	1:0:2:0	2	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Identify the problem given and design the algorithm using various algorithm design techniques.
- Implement various algorithms in a high level language.
- Analyze the performance of various algorithms.
- Compare the performance of different algorithms for same problem.

List of Experiments

Design, develop and implement the specified algorithms for the following problems using C/C++ Language in LINUX / Windows environment.

1. Sort a given set of elements using the Quicksort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.
2. Using Open MP, implement a parallelized Merge Sort algorithm to sort a given set of elements and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.
3. a. Obtain the Topological ordering of vertices in a given digraph.
   b. Compute the transitive closure of a given directed graph using Warshall's algorithm.
4. Implement 0/1 Knapsack problem using Dynamic Programming.
5. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
6. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm. Use virtual lab concept.
7. a. Print all the nodes reachable from a given starting node in a digraph using BFS method.
   b. Check whether a given graph is connected or not using DFS method.

8. Find a subset of a given set S = {sl, s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1, 2, 5, 6, 8} and d = 9 there are two solutions{1,2,6} and {1,8}. A suitable message is to be displayed if the given problem instance doesn't have a solution.
9. Implement any scheme to find the optimal solution for the Traveling Salesperson problem using dynamic programming.
10. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm. Use virtual lab concept.

Course Outcomes:

On completion of this course, the students are able to :

- Apply theoretical knowledge in practice through the execution of lab programmes.
- Learn the application and implementation of appropriate data structure in algorithm design.
- Identify the application of algorithms for real time problems.
- Differentiate between various algorithms performances.

Cloud Computing Foundations (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISI48	1:0:2:0	2	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Get exposed to cloud deployment and service models, cloud infrastructure, and the key considerations in migrating to cloud computing.
- Learn about the key considerations and steps involved in transitioning from the current state of their data center to a cloud computing environment.

Syllabus

Module - I

Journey to the Cloud: Drivers for cloud computing, cloud definition and characteristics, building cloud infrastructure – a phased approach from classic data center to virtual data center to Cloud, virtualization and its benefits. **05 Hours**

Module - II

Cloud Computing Primer: Cloud computing characteristics, cloud deployment models private, public, hybrid and community cloud, cloud services – SaaS, PaaS, and IaaS, cloud economics and challenges. **05 Hours**

Module - III

Cloud infrastructure and Management: Cloud infrastructure framework and components, infrastructure management and service creation tools, cloud service management processes asset and configuration management, service catalog management, financial management, capacity, performance and availability management, incident, problem and compliance management. **06 Hours**

Module - IV

Cloud Security: Basic information security concepts, cloud security concerns and threats, security mechanisms in cloud at compute, storage, and network layer, Governance, Risk and compliance in Cloud. **05 Hours**

Module - V

Cloud Migration Considerations: Considerations for choosing right application and cloud model, service provider specific considerations, cloud adoption phases, Financial and technical feasibility assessment, migration and optimization considerations.

05 Hours

Course Outcomes:

On completion of this course, the students are able to :

- Explain the phases of transition from classic data center to virtual data center and then to the Cloud.
- Explain the key characteristics, services, and deployment models of Cloud.
- Describe the Cloud infrastructure components and service management processes.
- Describe the Cloud security concerns and solutions.
- List the key considerations for migration to the Cloud.

Reference Books:

1. Cloud Computing Paperback – 2014, ISBN-13: 978-9332535923.
2. Cloud Computing for dummies.

Integrated Rural Development – Part 2

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
16ISH49	0:2:0:0	1	CIE:50 SEE:50	2 Hours	HSS

Course Objectives:

This course is an extension of the Integrated Rural Development course which was introduced in Semester 3. This course will extend the previous semester's work and will enable the students to:

- Continue working on the problems and challenges identified in the village.
- Apply their academic knowledge, talents, and abilities to come up with innovative and practical solutions to the challenges in the village.
- Foster a sense of entrepreneurship towards addressing the problems in the village.

Syllabus

Module - I

Overview: Overview of the course; summary of the experiences from previous semester with assigned mentors and supervisors; discussion of the challenges faced in the village identified previously. **03 Hours**

Module - II

Project Backlog Revision: Revisiting the challenges already identified in the previous semester and identifying possible project topics with the help of mentor and supervisor (this can be either continuation of the previous semester's project with a larger scope or a new project); student group discussion to finalize the new project definition; review of project definition with mentor and supervisor. **06 Hours**

Module - III

Project Plan Finalization: Modification of the previous semester's project plan to accommodate the new objectives; review of new proposaland plan with mentor and supervisor to finalize plan of work; distribution of workand needed resources and logistics within the group. **06 Hours**

Module - IV

Project Execution: Execution of the project as per the plan;conducting surveys to evaluate the impact of the project execution; collection ofproject deliverables; periodical review of the project execution status and collected artifacts (like aggregated data and survey reports) with mentor and supervisor. **10 Hours**

Module - V

Project Presentation: Creation of a final project report and a high-quality project presentation; both the project report and presentation should clearly articulate the motivation, how the project was conceptualized and executed, impact of the project, futuredirections in the project, and lessons learned by the students during the project; final review and evaluation by mentor and supervisor. **03 Hours**

Course Outcomes:

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

- Further develop their social and communication skills by interacting with residents of the village and within their team.
- Conceptualize long term solution to challenges in villages, thus developing a sense of entrepreneurship.
- Make an impact to rural sections of society, thus building their self-confidence.

Text Books:

1. Bhagawan Sri Sathya Sai Baba: "Service to Village is Service to God", Sri Sathya Sai Publications.

Reference Books:

1. Bhagawan Sri Sathya Sai Baba: "Man Management: A Value-Based Management Perspective", Sri Sathya Sai Publications.
2. Lt. Gen. M.L.Chibber: "Sai Baba's Mahavakya on Leadership : Book for Youth, Parents and Teachers."

E-Resources:

1. http://rural.nic.in/netrural/rural/index.aspx
2. www.annapoorna.org.in

ॐ ॐ ॐ ॐ ॐ

Program Educational Objectives (PEOs)

The graduates of Information Science and Engineering are expected to fulfill the following PEOs after a few years of their graduation.

PEO1	Pursue a successful career in the field of Information Science & Engineering or a related field utilizing his/her education and contribute to the profession as an excellent employee, or as an entrepreneur.
PEO2	Be able to work effectively in multidisciplinary environments and be responsible members/leaders of their communities.
PEO3	The graduates of Information Science and Engineering Program should be able to establish an understanding of professionalism, teamwork, ethics, public policy that allows them to become good professional Engineers.
PEO4	The graduates of Information Science and Engineering Program should be able to provide novel engineering solutions and efficient software designs with legal and ethical responsibility.
PEO5	Continuously improve by pursuing advanced degrees in engineering, business, or other professional fields through formal means or through informal self-study.

Program Outcomes (POs)

PO1	**Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and information science and engineering specialization to the solution of complex engineering problems.
PO2	**Problem Analysis:** Identify, formulate, review research literature, and analyze complex information science and engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	**Design/ Development of solutions:** Design solutions for complex information science and engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations
PO4	**Conduct Investigations of Complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5	**Modern Tool Usage:** Create, select, and apply appropriate techniques, resources, and modern information science and engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6	**The Engineer and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional information science and engineering practice.
PO7	**Environment and Sustainability:** Understand the impact of the professional information science and engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO8	**Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the information science and engineering practice.
PO9	**Individual and Team Work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	**Communication:** Communicate effectively on complex engineering activities with the information science and engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	**Project Management and Finance:** Demonstrate knowledge and understanding of the information science and engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	**Life-Long Learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of information science and technological change.