SUBJECT		Exam Duration	Contact Hours per Week			Credits	Max. Marks	Min Pass Marks
Code	Title		L	T	P			
CS-3401	Statistical Data Analysis using R	3 Hours	2	1	2	3 + 1	70	22

Course Description : This course provides an elementary introduction to probability and statistics with applications by using R programming language. This course provides a solid undergraduate foundation in both probability theory and statistics and at the same time provides an indication of the relevance and importance of the theory in solving practical problems in the real world with experiments using R programming.

Course Objectives : This course aims to provide an understanding of the basic concepts of R programming language and the basic concepts of statistics and probability with experimental approach using R. It will also focus on the random variable, different types of distributions, and estimation theory. Another objective of the course is to introduce the design of statistical hypothesis about the real world problem and to conduct appropriate test using R for drawing valid inference about the population characteristics.

UNIT - I :

Introduction to Statistics

Basics of Statistics : Types of data : qualitative vs. quantitative, Levels of measurement : nominal, ordinal, interval, ratio, Data collection methods.

Descriptive Statistics : Measures of central tendency : mean, median, mode, Measures of dispersion: range, variance, standard deviation.

UNIT - II :

Introduction to Probability : Basic probability concepts & rules, Conditional probability and independence, Bayes' theorem.

Discrete Distributions : Binomial distribution, Poisson distribution, Hypergeometric distribution.

Continuous Distributions : Normal distribution, Exponential distribution, Uniform distribution.

Sampling Distributions : Central Limit Theorem, Applications of sampling distributions.

UNIT - III :

Estimation : Point estimation and properties, Confidence intervals for means and proportions.

Basics of Hypothesis Testing : Null and alternative hypotheses, Type-I and Type-II errors, t-tests, chi-square tests, and ANOVA.

UNIT - IV :

Introduction to R language, Installing R and R Studio, Overview of R Studio, Working on the Console, Arithmetic Operators, Logical Operations, Using Functions. Data structures, variables, and data types in R: Creating Variables, Numeric, Character and Logical Data.

UNIT - V :

Data Vectors and Data Frames. Data visualization features in R: histograms, bar charts, pie charts, scatter plot. Application of statistical techniques to real-world data using R. R programming for hypothesis testing.

1 Hour Lecture (L) = 1 Credit 1 Hour Tutorial (T) = 1 Credit 2 Hours Practical (P) = 1 Credit

Contd. … 2

Course Outcomes :

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

- Understand the basic principles of statistics and probability
- Apply the principles of statistics and probability to real world problems.
- Explain probability distribution and solve problems.
- Explain the sampling, error and its applications.

Text Books :-

1. Sheldon M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", Academic Press, (2009).
2. P. Dalgaard. Introductory Statistics with R, 2nd Edition. (Springer 2008)
3. Maria Dolores Ugarte , Ana F. Militino , Alan T. Arnholt "Probability and Statistics with R" 2nd Edition on, CRC Press, 2016.

Reference Books :-

1. D.C. Montgomery and G.C. Runger, "Applied Statistics and Probability for Engineers", 5th edition, John Wiley & Sons, (2009).
2. Robert H. Shumway and David S. Stoffer, "Time Series Analysis and Its Applications with R Examples", Third edition, Springer Texts in Statistics, (2006).
3. Michael Akritas, " Probability & Statistics with R for Engineers and Scientists", 2nd Edition on, CRC Press, 2016.

Suggested List of Experiments :-

(1) Installation of R in Windows and Linux environment.
(2) Write R program to find measures of central tendency.
(3) Write R program to store data into a List and perform different operations.
(4) Write R program to store data into Data frame and perform different operations.
(5) Write R program to find sum of elements of vector.
(6) Write R program to find mean, variance, standard deviation for the given discrete probability distribution.
(7) Write R program to find mean, variance, standard deviation for the given continuous probability distribution.
(8) Write R program to represent the given data in the form of graphs using built in functions.
(9) Write R program to fit Binomial distribution to the given data.
(10) Write R program to fit Poisson distribution to the given data.
(11) Write R program for Z test.
(12) Write R program for t test.
(13) Write R program for F test.
(14) Write R program for Chi-square test.

*****

SUBJECT		Exam Duration	Contact Hours per Week			Credits	Max. Marks	Min Pass Marks
Code	Title		L	T	P			
CS-3402	Data Structures	3 Hours	2	1	2	3 + 1	70	22

Course Description : Study of advanced programming topics focused on logical structures of data. Data structure operations. Topics include linked lists, stacks, queues, trees, graphs. Also covers searching, and sorting techniques. Operations and Applications of: Linked Lists; Stacks and Queues; Trees, Search trees and Heaps; Multi-way Trees and Graphs.

Course Objectives :

- To learn the fundamentals of Data Structures, features and applications.
- To learn the Principles of Data Structures.
- To understand and gain knowledge on Linear and Non-linear Data Structures.
- To apply different searching and sorting techniques.
- To analyze and identify suitable data structure for computational problem solving.

UNIT - I :

Introduction : Introduction to Data Structures, Array, Structure, Pointer and Function.

Linked Lists : Linked List definition, Singly Linked Lists, Circular Linked List, Doubly Linked List, Circular Doubly Linked List, Application of Linked Lists.

UNIT - II :

Stacks and Queues :

Stack : Stack operations, Array and Linked List based stack implementation. Stack applications : Prefix, Infix, and Postfix expressions.
Queues : Queue operations, Array and Linked List based Queue implementation, Queue applications.

UNIT - III :

Trees, Search Trees and Heaps :

Trees : Tree concepts and type of trees, Binary Trees. Binary Search Trees (BST) : Basic concepts, BST operations, BST applications. AVL Search Trees : Basic concepts, AVL Tree implementations. Threaded binary tree and Heap tree.

UNIT - IV :

Multiway Trees and Graphs :

Multiway Trees : B-Trees, Simplified B-Trees, B-Tree variations.
Graphs : Basic concepts; Graph storage structures : Adjacency matrix, Adjacency list; Graph Traversals : Depth-first traversal, Breadth-first traversal. Prim's and Kruskal's algorithm for Spanning Tree, Dijkstra's Shortest Path algorithm.

UNIT - V :

Searching and Sorting :

Searching : Linear search and Binary search.
Sorting : Internal and External sorting. Various Sorting Methods : Insertion Sort, Selection sort, Bubblel Sort, Quick Sort, Merge Sort, Heap Sort.

1 Hour Lecture (L) = 1 Credit 1 Hour Tutorial (T) = 1 Credit 2 Hours Practical (P) = 1 Credit

Contd. … 2

Course Outcomes :

At the end of the course, the student will :

- Design solutions for engineering problems using linear and non-linear data structures.
- Develop solutions for Complex computational problems.
- Apply appropriate data structure to provide solutions for real time problems.
- Apply contextual knowledge of data structures to design applications for societal applications like payroll systems, web applications, banking and financial systems.

Text Books :-

1. Richard Gileberg and Behrouz A. Forouzan, Data Structures : A Pseudocode Approach with C, Cengage Learning, Second Edition, 2007.
2. E. Balagurusamy, "Data Structures Using C", Tata McGraw Hill, 2013.
3. E. Horowitz and S. Sahni, "Fundamentals of Data Structures", Publisher Computer Science Press, Second Edition, 2008.
4. R.L. Kruse, "Data Structure and Program Design", Prentice Hall, Second Edition, 1996.

Reference Books :-

1. Debasis Samanta, Classic Data Structures, PHI Learning, Second Edition, 2009.
2. Aaron M. Tenenbaum, Yedidyah Langsam, and Moshe J. Augenstein, Data Structures Using C, Pearson Education, 2005.
3. G.A.V. Pai, "Data Structures and Algorithms", Tata Mc-Graw Hill, Second Edition, 2009.

Suggested List of Experiments :-

(1) Write a C program to find the standard deviation of data elements stored in array.
(2) Write a C program that uses functions to perform the following :-
  (a) Create a singly linked list of integers.
  (b) Delete a given integer from the above linked list.
  (c) Display the contents of the above list after deletion.
(3) Write a C program that uses functions to perform the following :-
  (a) Create a doubly linked list of integers.
  (b) Delete a given integer from the above doubly linked list.
  (c) Display the contents of the above list after deletion.
(4) Write a C program to find the largest element in a given doubly linked list.
(5) Write a C program to implement stack using array that uses functions to perform the following :-
  (a) Convert a given infix expression into its postfix equivalent.
  (b) Evaluate a given postfix expression
(6) Write a C program to implement stack using linked list.
(7) Write a C program that uses functions to implement circular queue using array.
(8) Create a binary search tree of characters and traverse it recursively in Pre-order, Post-order and In-order.
(9) Write C programs for implementing the following graph traversal algorithms :-
  (a) Depth first traversal (b) Breadth first traversal.
(10) Write C programs for implementing the following sorting methods to arrange a list of integers in ascending order : (a) Insertion sort (b) Merge sort (c) Quick sort (d) Selection sort (e)Bubble sort.
(11) Write C programs for implementing the following searching methods to find an element in a list of integers : (a) Linear Search (b) Binary Search.

These are the General List of Experiments & the list of Experiments may vary as per course Instructor.

*****

SUBJECT		Exam Duration	Contact Hours per Week			Credits	Max. Marks	Min Pass Marks
Code	Title		L	T	P			
CS-3403	Discrete Structures	3 Hours	3	1	0	4 + 0	70	22

Course Description : To identify the basic properties of Logic, Relation and Function, graphs and trees to model simple applications and to get familiar and understand the fundamental notions in discrete structure. Distinguish between the notion of discrete and continuous mathematical structures.

Course Objectives :

- To get familiar and understand the fundamental notions in discrete mathematics.
- To understand and demonstrate the basic concept of an algorithm and its application in combinatorial mathematics.
- To identify the basic properties of graphs and trees and model simple applications

UNIT - I :

Set Theory : Definition of sets, countable and uncountable sets, Venn Diagrams.
Relation : Definition, types of relation, composition of relations, Pictorial representation of relation, Equivalence relation, Partial ordering relation.
Function: Definition, type of function:, one to one, into and onto function, inverse function, composition of functions, recursively defined functions. Mathematical induction: theorem proving technique. Pigeon-hole principle.

UNIT - II :

Algebraic Structures : Definition, Properties, types: Semi Groups, Monoid, Groups, Abelian group, properties of groups. Subgroup, cyclic groups, Cosets, factor group, Permutation groups. Normal subgroup, Homomorphism and isomorphism of Groups, Rings and Fields: definition and standard results.

UNIT - III :

Propositional Logic : Proposition, First order logic. Basic logical operation, truth tables, tautologies, Contradictions, Algebra of Proposition, logical implications, logical equivalence, predicates, Normal Forms. Introduction to finite state machine Finite state machines as models of physical system equivalence machines.

UNIT - IV :

Graph Theory : Introduction and basic terminology of graphs, Planer graphs, Multigraphs and weighted graphs, Isomorphic graphs, Paths, Cycles and connectivity, Shortest path in weighted graph, Introduction to Eulerian paths and circuits. Hamiltonian paths and circuits, Graph colouring, chromatic number, Isomorphism and Homomorphism of graphs.

UNIT - V :

Posets, Hasse Diagram and Lattices : Introduction, ordered set, Hasse diagram of partially. ordered set, isomorphic ordered set, well ordered set, properties of lattices, Bounded and Complemented lattices, Combinatories. Introduction to Recurrence Relation and Recursive algorithms.

1 Hour Lecture (L) = 1 Credit 1 Hour Tutorial (T) = 1 Credit 2 Hours Practical (P) = 1 Credit

Contd. … 2

Course Outcomes :

At the end of the course, the student will :

- Understand sets, relations, functions and discrete mathematical structures.
- Apply Propositional logic and first order logic to solve problems.
- Formulate and solve tree and graph problems.
- Formulate and Count discrete event occurrences and solve recurrence relations.

Text Books :-

1. C.L. Liu And D.P. Mahapatra," Elements Of Discrete Mathematics : A Computer Oriented Approach", McGraw Hill, Third Edition, 2012.
2. Kenneth H. Rosen, "Discrete Mathematics And Its Applications" McGraw Hill, Seventh Edition, 2012 (Indian Adaptation By Kamala Krithivasan, lit Madras).

Reference Books :-

1. R. Balakrishnan and K. Ranganathan, "A Text Book Of Graph Theory". Springer.
2. Thomas Koshy, "Discrete Mathematics with Applications", Elsevier, 2009.
3. Gary Haggard, John Schlipf and Sue Whitesides, "Discrete Mathematics for Computer Science", Cengage Learning Publisher, 2005.
4. B. Bollobás, "Modern Graph Theory", Springer, New York 1998.

*****

SUBJECT		Exam Duration	Contact Hours per Week			Credits	Max. Marks	Min Pass Marks
Code	Title		L	T	P			
CS-3404	Software Development using JAVA	3 Hours	2	1	2	3 + 1	70	22

Course Description : Java is a platform-independent object-oriented programming language used to create stand-alone applications and applets for the World Wide Web. This course gives the student a basic understanding of the Java language and its role in the Object Oriented World. The student creates simple applications and applets.

Course Objectives : This course aims to :-

- Gain knowledge about basic Java language syntax and semantics to write Java programs and use concepts such as variables, conditional and iterative execution methods, constructor etc.
- Understand the fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc and exception handling mechanisms.
- Understand the principles of inheritance, packages and interfaces.
- Develop front end of an application using AWT.
- Develop applications using JSP and Applets.

UNIT - I :

Introduction to Java Classes and Objects :
Java features : Java syntax, data types, data type conversions, control statements, operators and their precedence. **Introduction to Class :** Instance members and member functions, String Handling. **Wrapper classes :** Arrays and Vectors.

UNIT - II :

Inheritance and Polymorphism : Class relationships : Inheritance and its types, merits and demerits. Association inheritance. Polymorphism : Dynamic method dispatch, Compile time and Runtime polymorphism, Abstract classes. Interfaces and packages.

UNIT - III :

Exception Handling : Need for exceptions, Checked Unchecked exceptions, creating exceptions. Types of Java Exceptions, Methods of Exception Handling: The Try and Catch Block, throw Statement, Finally Block, Java Throw v/s Throws.

UNIT - IV :

Multithreading & Java I/O : Multithreading: Introduction of Thread & Multithreading, Java thread class, Life Cycle of thread, Java Thread Method, Priorities and scheduling. Thread Synchronization and its life cycle. Basic concept of streams: I/O stream & reader-writer classes. File handling.

UNIT - V :

Applets & JSP : Applet and its Life Cycle, Event Delegation Model and Java event handling, Windows Event, Introduction of Java Server Page (JSP), features of JSP & creating simple JSP Page.
AWT Components : Labels, Buttons, Check boxes, Radio Button, Text Field and Text Area. A Brief overview of JDBC process and JDBC driver.

1 Hour Lecture (L) = 1 Credit 1 Hour Tutorial (T) = 1 Credit 2 Hours Practical (P) = 1 Credit

Contd. … 2

Course Outcomes :

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

- Understand basic concepts of object oriented programming.
- Identify classes, objects, members of a class and relationships among them needed for a specific problem.
- Demonstrate the concepts of polymorphism and inheritance.
- Understand various error handling techniques using exception handling. Apply the knowledge gained for their project work as well as to develop some GUI applications.
- Understand basic concepts of software development using Java.

Text Books :-

1. Let us java, by Yashvant Kanetkar, 9th Edition.
2. The complete Reference, by Harbert Schildt, Tata McGrow Hill, 7th Edition.
3. Object Oriented Programming. by Timothy, Budd, Pearson Education, 2nd Edition.

Reference Books :-

1. Core JAVA Vol-1, by Cay S. Horstmann, Pearson Education, 9th Edition.
2. The Object Primer, by Scott W Amber, Cambridge, 3rd Edition.
3. Head First Java, by Kathy Sierra, Bert Bates, 2nd Edition.

Suggested List of Experiments :-

(1) Lab Setup and Introduction to Java Fundamentals.
(2) Create a class named 'Student' with String variable 'name' integer variable 'roll_no' to store roll number of student integer variable Java' to store marks for Java subject and maths to store marks for Maths subject. Assign the value of roll no as '2" and that f name as your own name (e.g. Bharti), marks for java and maths 80, 90 respectively by creating object of the class Student.
(3) Create a class named 'Bank Account' with String variable 'name' integer variable 'balance' and 'account number' A person can deposit and withdraw money from account and can get the status of balance. Write a java program to initialize data using constructors.
(4) Write a program to create a 'Room' class, the attributes of this class is room no, room- type, room area. In this class the member functions are set data and display data. Use parameterized constructors to initialize data members.
(5) A class named "Arithmetic' with a method named 'add' that takes integers as parameters and returns an integer denoting their sum. A class named 'Adder' that inherits from a super class named Arithmetic. Your classes should not be public.
(6) Write a java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Tri-angle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape
(7) Write a program to implement multiple inheritances using interface in java.
(8) Write a program to implement Abstract Class
(9) Write a program to implement concept of packages :-
    (a) Same package - No subclasses (b) Same package – Subclasses
    (c) Different package - No subclasses (d) Different package - Subclasses
(10) Write a program that creates a user interface to perform integer division. The user enters two numbers Num1 and Num2. If Num1 and Num2 were not integers, the program would throw a Number Format Exception. If Num2 were zero, the program would throw an Arithmetic Exception Display the exception message.
(11) Write a program to show "HELLO JAVA" in Explorer using Applet.
(12) Write a java program that implements a multi-thread application that has three threads: First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
(13) Creating simple JSP PAGE.
(14) Creating GUI using AWT components.

These are the General List of Experiments & the list of Experiments may vary as per course Instructor.

*****

SUBJECT		Exam Duration	Contact Hours per Week			Credits	Max. Marks	Min Pass Marks
Code	Title		L	T	P			
CS-3405	Web Development Lab	4 Hours	0	0	4	0 + 2	30	10

Course Description : On completion of this course, the student will be familiar with client server architecture and able to develop a web application using java technologies. Students will gain the skills and project-based experience needed for entry into web application and development careers.

UNIT - I :

Introduction to internet : History of Internet, World Wide Web, Internet Addressing, Browser, URL, Web server, Website, page design, homepage layout, design concept, Domain Name. Basic concept of Cookies and Sessions.

UNIT - II :

HTML : Basic structure of an HTML document, Creating an HTML document, HTML Tags and Attributes, HTML Basic Tags, Formatting Tags, HTML Color Coding, Table Tag, Div and Span Tag for grouping. List : Unordered lists, ordered list, definition list. Image : Image and Image mapping. Working with Text, Hyperlinks, Forms and controls.

UNIT - III :

CSS : Concept of Cascading Style Sheet (CSS), CSS Syntax, Types of CSS, CSS Properties, CSS Styling (Background, Text Format, Controlling Fonts), CSS Id and Class, Box Model (Introduction, Border properties, Padding Properties, Margin properties).
XML : Rules for Writing XML. Elements, Attributes, and Values. XSL and XSLT, Transforming XML with XSLT, Working with DTDs.

UNIT - IV :

Java Script : Introduction to JavaScript, Script Writing Basics, Enhancing HTML Documents with JavaScript, JavaScript Engines, Values, Variable and Operator, Operator Precedence, JavaScript Objects, Introduction of conditional statements, array, loop and function in JavaScript.

UNIT - V :

PHP : Basics of PHP, Evaluation of PHP, Basic Syntax, Defining variable and constant, PHP Data type, Operator and Expression. Conditional statements, array, loop and function in PHP.

Reference Books :-

1. Jon Duckett "Beginning Web Programming" WROX.
2. Sebesta, "Programming world wide web" Pearson.
3. Dietel and Nieto, "Internet and World Wide Web-How to program", PHI/Pearson Education Asia.
4. Wang, "An Introduction to web Design and Programming". Thomson

1 Hour Lecture (L) = 1 Credit 1 Hour Tutorial (T) = 1 Credit 2 Hours Practical (P) = 1 Credit

Contd. … 2

List of Experiments :-

(1) Design web pages for your college containing a description of the courses, departments, faculties, library etc, use href, list tags.
(2) Create your class timetable using table tag. Create user Student feedback form (use textbox, text area, checkbox, radio button, select box etc.)
(3) Write html code to develop a web page having two frames that divide the webpage into two equal rows and then divide the row into equal columns fill each frame with a different background color.
(4) Create your resume using HTML tags also experiment with colors, text, link, size and also other tags you studied.
(5) Design a web page of your home town with an attractive background color, text color, an Image, font etc. (use internal CSS), Use Inline CSS to format your resume that you created.
(6) Use External CSS to format your class timetable as you created. Use External, Internal, and Inline CSS to format college web page that you created.
(7) Develop a JavaScript to display today's date. Develop simple calculator for addition, subtraction, multiplication and division operation using JavaScript.
(8) Create HTML. Page with JavaScript which takes Integer number as input and tells whether the number is ODD or EVEN.
(9) Create HTML Page that contains form with fields Name, Email, Mobile No. Gender, Favorite Color and a button now write a JavaScript code to combine and display the information in the textbox when the button is clicked.
(10) Implement Validation in above Feedback Form. Use regular expression for validation in Feedback Form. Using ajax retrieve data from a TXT file and display it.
(11) Create XML file to store student information like Enrollment Number, Name, Mobile Number, Email Id. Create XSL file to convert above XML file into XHTML file.
(12) Create DTD for above XML. File, Create XML Schema for above.
(13) Write a php program to display today's date in dd-mm-yyyy format.
(14) Write a php program to check if number is prime or not.
(15) Write a php program to print first 10 Fibonacci Numbers.
(16) Create HTML. page that contain textbox, submit/reset button. Write php program to display this information and also store into text file.
(17) Write a php script to read data from txt file and display it in html table (the file contains info in format Name: Password: Email).
(18) Write a PHP Script for login authentication. Design an html form which takes user name and password from user and validate against stored username and password in file.
(19) Design an HTML page which takes Name, Address, Email and Mobile.

These are the General List of Experiments & the list of Experiments may vary as per course Instructor.

*****

SUBJECT		Exam Duration	Contact Hours per Week			Credits	Max. Marks	Min Pass Marks
Code	Title		L	T	P			
CS-3406	Computer Workshop	2 Hours	0	0	2	0 + 1	-	-

Course Description : The main objective of this course is to provide an overview of the basic functions of personal computers and instructs students in how to use them. It is designed for persons who have little or no experience with computers. Its purpose is to make students comfortable with personal computers and to provide the basic knowledge and skills needed to perform the major computer functions without assistance. These skills will be applied to the basics of applications such as word processing, spreadsheets (tables of numbers), email and the internet.

Course Objectives : The objectives of this course is to make the students to :

- Understand the basic computer hardware and system requirements.
- Make them to install Operating system and operate basic software utilities.
- Familiar with the use of websites and access search engines to find information and troubleshoot basic computer problems.
- Realize the importance of basic technologies related to an office environment.

UNIT - I :

Computer Peripherals : Input and Output Devices, Primary Component, Computers Language, Serial and parallel Communication, SMPS.

UNIT - II :

Motherboard : Type, form factor, BUS, IRQ, Chipset, I/O Ports and Connectors, CMOS. Memory: Type, Memory Modules, Development of RAM.

UNIT - III :

Networking : Introduction, Types of Networks, Topology, protocols and Ports. Networking devices like Routers, Switches, Hub, Repeater, NIC Cards, Bridge. Networking Media : Wire, Wireless, Cables, Crimping, UTP.

UNIT - IV :

Operating Systems : Edition, Requirement, Types of Installation, Driver Installation, Generation of processors, Study of various ports, Steps and precautions to assemble computer.

UNIT - V :

Data base management overview, Excel : Environment, entering and editing data, modifying of worksheet, functions, formatting.

Course Outcomes : At the end of the course, the student will be able to :

- Explain the working principles of computer peripherals.
- Identify, analyze and apply the troubleshooting techniques to solve operating system and hardware problems.
- Create/manipulate the documents, spreadsheets and presentations using MS office and web pages using HTML for real-time applications.

1 Hour Lecture (L) = 1 Credit 1 Hour Tutorial (T) = 1 Credit 2 Hours Practical (P) = 1 Credit

Contd. … 2

Reference Books :-

1. Hardware Bible by Winn L. Rosch
2. Hardware and Software of Personal Computers by Sanjay K. Bose.
3. Fundamentals of Computers by V. Rajaraman.
4. Computer Studies - A first course by John Shelley and Roger Hunt.
5. Computer Fundamentals, MS Office and Internet & Web Technology by Dinesh Maidasani.
6. Modern Computer Hardware Course by M Lotia, P Nair, P Lotia

List of Experiments :-

(1) Study of peripherals of a computer, components in a CPU and its functions.
(2) Assembling and disassembling of PC.
(3) Installation of Operating Systems–Windows.
(4) Installation of Operating Systems–LINUX.
(5) Hardware Troubleshooting.
(6) Software Troubleshooting.
(7) Providing Internet connectivity.
(8) Understand modern application development.
(9) Configuring Firewalls and installation of Antivirus software.
(10) Introduction to office automation software.

*****

SUBJECT		Exam Duration	Contact Hours per Week			Credits	Max. Marks	Min Pass Marks
Code	Title		L	T	P			
EC-3491	Digital Systems	3 Hours	2	1	2	3 + 1	70	22

Course Description : This course introduces the basic logic functions components and methodologies used in the design of digital systems Digital electronic topics will include the basic logic gates, Boolean algebra, number systems, digital arithmetic, combinational logic circuits, multiplexers, decoders and flip-flops and registers. Digital system applications will include counters, magnitude comparators, Analog-to-Digital and Digital-to-Analog conversions Clocks and synchronization CMOS, PMOS logic circuits Combinational logic design, TDM, sampling theorem, PCM, introduction to BPSK and BFSKshannon's theorem for channel capacity, IC 555 and multivibrators.

Course Objectives : Students will try to learn:

- To understand number representation and conversion between different representation in digital electronic circuits.
- To analyze logic processes and implement logical operations using combinational logic circuits.
- To understand characteristics of memory and their classification.
- To understand concepts of sequential circuits and to analyze sequential systems in terms of state machines.
- To understand concept of Programmable Devices, PLA. PAL.

UNIT - I :

Review of Number Systems and Number Base Conversions : Binary codes, Boolean algebra. Boolean functions, Logic gates. Simplification of Boolean functions, Karnaugh map methods, SOP-POS simplification, NAND-NOR implementation.

UNIT - II :

Combinational Logic : Half adder, Hall Subtractor, Full adder. Full Subtractor, look-ahead carry generator, BCD adder, Series and parallel addition, Multiplexer - de-multiplexer, encoder- decoder, arithmetic circuits, ALU

UNIT - III :

Sequential Logic : flip flops, D.T. S-R, J-K Master-Slave, racing condition, Edge & Level triggered circuits, Shift registers: Asynchronous and synchronous counters, their types and state diagrams. Semiconductor memories, Introduction to digital ICs 2716, 2732 etc. and their address decoding. Modern trends in semiconductor memories such as DRAM, FLASH RAM etc. Designing with ROM and PLA.

UNIT - IV :

Introduction to A/D & D/A convertors & their types, sample and hold circuits, Voltage to Frequency & Frequency to Voltage conversion. Multivabrator Bistable, Monostable. Astable, Schmitt trigger, IC 555 & Its applications. TTL, PMOS, CMOS and NMOS logic. Interfacing between TTL to MOS.

UNIT - V :

Introduction to Digital Communication: Nyquist sampling theorem, time division multiplexing. PCM, quantization error, introduction to BPSK & BFSK modulation schemes. Shannon's theorem for channel capacity.

1 Hour Lecture (L) = 1 Credit 1 Hour Tutorial (T) = 1 Credit 2 Hours Practical (P) = 1 Credit

Contd. … 2

Course Outcomes :

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

- Develop a digital logic and apply it to solve real life problems.
- Analyze, design and implement combinational logic circuits.
- Classify different semiconductor memories.
- Analyze, design and implement sequential logic circuits.

Reference Books :-

1. Morris Mano, Digital Circuits & Logic Design, PHI.
2. Gothman, Digital Electronics, PHI.
3. Tocci, Digital Electronics, PHI.
4. Mavino & Leach, Digital Principles & Applications, PHI.
5. Taub and schilling, Digital Integrated electronics.
6. Simon Haykin, Introduction to Analog & Digital Communication, Wiley.
7. Lathi B.P., Modern analog& digital communication, Oxford University.

List of Experiments :-

(1) To study & verify different types of logic gates.
(2) To verify the De-Morgan's theorem and Boolean rules.
(3) To design the logic circuit of Even and Odd Parity Generator and Checker.
(4) To study & design logic circuit for Half Adder and Full Adder.
(5) To study & design logic circuit for Half Subtractor and Full Subtractor.
(6) To study & verify the different types of Flip-Flop.
(7) To study & verify the MUX and DE-MUX.
(8) To study & verify the different Multivibrator.
(9) To study & verify the Full Adder using 4x1 and 8x1 multiplexer.
(10) To study & verify IC 555 Timer.

These are the General List of Experiments & the list of Experiments may vary as per course Instructor.

*****