Syllabus Content

Semester – III

Structure

S.No.	**Course Code**	**Course Title**	**Category**	**L**	**T**	**P**	**Credits**
1	MAC 231	Mathematics-III	FC(BS)	3	1	0	4
2	MAC 232	Discrete Mathematics	FC(CS)	3	0	0	3
3	CSC 233	Object Oriented Programming with Java	FC(CS)	3	0	0	3
4	ECC 234	Digital Circuits and Systems	FC(OE)	3	0	0	3
5	HSC 235	Engineering Economics and Costing	FC(HS)	3	0	0	3
6	CSC 236	Computer Organization and Architecture	FC(CS)	3	0	0	3
7	CSL 237	Object Oriented Programming with Java Lab.	FC(CS)	0	0	3	2
8	ECL 238	Digital Circuits Lab.	FC(OE)	0	0	3	2
				Total Credit :			**23**

<table>
<tr><th colspan="9">MATHEMATICS-III</th></tr>
<tr><td><b>Course Code</b></td><td><b>MAC 231</b></td><td><b>L-P-T-Cr.:</b></td><td><b>3</b></td><td><b>0</b></td><td><b>1</b></td><td><b>4</b></td><td><b>Semester:</b></td><td><b>III</b></td></tr>
<tr><td><b>Category:</b></td><td colspan="8">FC(BS)</td></tr>
<tr><td><b>Course Objectives:</b></td><td colspan="8">To introduce the concept of probability and statistics and their implementation in real life situations. To give a foundation of complex function and their approach to different types of series & integration.</td></tr>
<tr><td><b>UNIT – I:</b></td><td colspan="7"><b>PROBABILITY</b></td><td><b>(10 Hours)</b></td></tr>
<tr><td colspan="9">Probability: Sample space and events – The axioms of probability – some elementary theorems – conditional probability – Bayes’ theorem. Random variables – discrete and continuous distribution – distribution functions-Binomial, poison and Normal distribution- sampling distribution – population and samples – proportions, sums and differences.</td></tr>
<tr><td><b>UNIT – II:</b></td><td colspan="7"><b>STATISTICS</b></td><td><b>(10 Hours)</b></td></tr>
<tr><td colspan="9">Estimations: Point estimation – interval estimation – Bayesian estimation. Testing of hypothesis: means – hypothesis concerning one or two means – Type I and Type II errors. One tail, tow-tail tests. Test of significance – student’s t- test. F-test, test. Estimation of proportion.</td></tr>
<tr><td><b>UNIT – III:</b></td><td colspan="7"><b>COMPLEX FUNCTIONS</b></td><td><b>(10 Hours)</b></td></tr>
<tr><td colspan="9">Functions of complex variable – Continuity – Differentiability – Analyticity – Properties – Cauchy- Riemann equations in Cartesian and polar coordinates. Harmonic and conjugate harmonic functions – Milne – Thompson Method.</td></tr>
<tr><td><b>UNIT – IV:</b></td><td colspan="7"><b>COMPLEX INTEGRATION</b></td><td><b>(10 Hours)</b></td></tr>
<tr><td colspan="9">Line integral – evaluation along path and by definite integration – Cauchy’s integral theorem – Cauchy’s integral formula – Taylor’s series expansion- singularities (isolated, pole, essential) – Residues – evaluation of residue by Laurent series. Residue theorem. Evaluation of integrals of different type.</td></tr>
<tr><td colspan="9"><b>TEXT BOOKS:</b></td></tr>
<tr><td>(1)</td><td colspan="8">Introduction to Probability and Statistics by William Mendenhall, Cengage learning.</td></tr>
<tr><td>(2)</td><td colspan="8">Higher Engineering Mathematics by B.V. Ramana (Tata McGraw-Hill)</td></tr>
<tr><td colspan="9"><b>REFERENCE BOOKS:</b></td></tr>
<tr><td>(1)</td><td colspan="8">Advanced Engineering Mathematics, EriwinKreyszig’s 8th Edition. Wiley Indian Publisher.</td></tr>
<tr><td>(2)</td><td colspan="8">Advance Engineering Mathematics by Jain and S.R.K. Iyengar, Narosa Publications</td></tr>
<tr><td rowspan="2"><b>Course Outcomes:</b></td><td colspan="8">After completion of this course the students should be able to:</td></tr>
<tr><td colspan="8">The probability models and statistical methods give a pro forma to analyze the data in various scientific disciplines which increase their research interests in a basic level.
The details of complex function theory give a smooth entry to many technical and bio engineering fields in an analytical way.</td></tr>
</table>

DISCRETE MATHEMATICS

Course Code	**MAC 232**	**L-P-T-Cr.:**	**3**	**0**	**0**	**3**	**Semester:**	**III**
Category:	FC(BS)							
Course Objectives:	The objective is to introduce Logic, Graphs and Algebraic structures.							

UNIT – I:	**LOGIC**	**(10 Hours)**

Mathematical reasoning; propositions; negation disjunction and conjunction; implication and equivalence; normal form; truth tables; predicates; quantifiers; natural deduction; rules of Inference; methods of proofs; resolution principle; Automatic theorem proving, Fuzzy logic: fuzzy relation, pattern classification, fuzzy analysis, distance between fuzzy sets, area perimeter, height, width of fuzzy subsets.

UNIT – II:	**SETS, RELATION & FUNCTIONS**	**(10 Hours)**

Set theory; Paradoxes in set theory; inductive definition of sets and proof by induction; Peono postulates; Relations; representation of relations by graphs, Warshall’s algorithm; properties of relations; equivalence relations and partitions; Partial orderings; Posets; Linear and well-ordered sets; Functions; mappings; injection and surjections; composition of functions; inverse functions; special functions; pigeonhole principle.

UNIT – III:	**GRAPH THEORY**	**(10 Hours)**

Graphs: representation of Graphs, operations on graphs, paths and circuits, graph traversals, shortest path in weighted graphs, Eulerian paths and circuits, Hamiltonian paths and circuits, Travelling sales persons problem, Planar graphs, Graph Coloring, Application of Graphs, Tress: Rooted trees, Binary search trees, Spanning trees, Minimum spanning trees, Kruskal’s Algorithm, Prims Algorithm.

UNIT – IV:	**ALGEBRAIC STRUCTURES**	**(10 Hours)**

Groups and rings: Semigroups, monoids, groups and subgroups, Cosets and Lagrange’s theorem, Codes and Group codes, applications of groups to error detection and correction. Boolean Algebras: Lattices and algebraic systems, Principle of duality, Distributive and complemented lattices, Boolean functions and Boolean expressions, Simplification of logic expressions using Karnaugh Map, Simplification of logic expressions using Quine-McClusky method.

TEXT BOOKS:

(1)	C. L. Liu, Elements of Discrete Mathematics, McGraw-Hill.
(2)	K. H. Rosen, Discrete Mathematics and applications, TataMcGraw Hill

REFERENCE BOOKS:

(1)	*J .L. Mott, A. Kandel, T.P .Baker*, Discrete Mathematics for Computer Scientists and Mathematicians, second edition 1986, Prentice Hall of India.
(2)	*R. Grimaldi and B V Ramana*, Discrete and combinatorial mathematics: An applied introduction, Pearson education.
(3)	*S. Lipschutz*, Discrete Mathematics, McGraw Hill Education (India) Pvt. Ltd., 2005.
(4)	*J. P. Tremblay and R. Manohar*, Discrete Mathematical Structures with Applications to Computer Science, McGraw Hill Education (India) Pvt. Ltd., 2001.
(5)	*B. Kolman and R. C. Busby*, Discrete Mathematical Structures for Computer Science, Prentice Hall of India, 5th Edition, 2002.
(6)	*N. Deo*, Graph Theory with applications to Engineering & Computer Science, Prentice Hall of India, 2006.

Course Outcomes:	After completion of this course the students should be able to:
	Logics and graphs are the key points for algorithm, Networking, coding and many more recent areas. This course helps to understand some areas of computer science in detail.

OBJECT ORIENTED PROGRAMMING WITH JAVA

Course Code	**CSC233**	**L-P-T-Cr.:**	**3**	**0**	**0**	**3**	**Semester:**	**III**
Category:	PC(CS)							

Course Objectives:	(1)	Learn the concepts of object-oriented programming.
	(2)	Introduce the implementation of inheritance, packages and interfaces.
	(3)	Understand the concepts of exception handling and multithreading.
	(4)	Introduce the java collection framework and I/O classes.

UNIT – I:		**(10 Hours)**

Java Evolution and Environment: Java evolution, overview of java language, java history, features of java, how java differs from C and C++, java and World Wide Web, web browser.

Java Environment: Java Development Kit (JDK), Application Programming Interface (API), java programming structure, java tokens, constants, variables, expressions, decision making statements and looping, java statements, overview of arrays and strings, machine neutral, Java Virtual Machine (JVM), Command Line Arguments.

Arrays and Strings: One-dimensional arrays, creating an array, declaration of arrays, initialization of arrays, two-dimensional arrays, string arrays, string methods, string buffer class, vectors, wrapper classes, Basic I/O Streams: Scanner, buffered reader.

UNIT – II:		**(10 Hours)**

Classes, Objects and Methods: Introduction, defining a class, creating objects, accessing class members, constructors, method overloading, staticmembers. Inheritance: Defining a sub-class, sub-class constructor, multi-level variables, final classes and finalize methods, abstract methods and classes, visibility control.

Managing Errors and Exceptions: Introduction, types of errors: compile time and run-time errors, exceptions, types of exceptions, syntax of exception handling code, multiple catch statements, using finally statement, throwing our own exceptions.

UNIT – III:		**(10 Hours)**

Interfaces, Package and Multi-threaded Programming: Introduction, defining interfaces, extended interfaces, implementing interfaces. Package: Creation, importing a package and user-defined package. Threads: Introduction to threads, creating threads, extending the thread class, implementing the ‘runnable’ interface, life-cycle of a thread, priority of a thread, synchronization, and deadlock

UNIT – IV:		**(10 Hours)**

Applet programming: Introduction, how applets differ from applications, building applet code, applet life cycle, about HTML, designing a web page, passing parameters to applets, getting input from the user. Graphics Programming: Introduction, abstract window toolkit class hierarchy, frames, event-driven programming, layout managers, panels, canvases, drawing geometric figures. Introduction to Swings: Introduction to Swings, overview of Swing components: Jbutton, JCheckBox, JRadioButton, JLabel, JTextField, JTextArea, JList.

Introduction to Networking: InetAddress class, socket class, URL class.

TEXT BOOKS:

(1)	Herbert Schildt, The Java Complete References, 9/e, Tata McGraw Hill, 2014.

REFERENCE BOOKS:

(1)	Y.Daniel Liang, An Introduction to JAVA Programming, Tata McGraw Hill, 2009.
(2)	Kathy Sierra, Head First java, 2/e, Shroff Publishers, 2012.
(3)	E. Balaguruswamy, Programming with JAVA, 2/e, Tata McGraw Hill, 2014.

Course Outcomes:	After completion of this course the students should be able to:	
	(1)	Understand the concepts and implement arrays and strings.
	(2)	Understand the object-oriented programming concepts, solve real world problems and implement the concepts of exception Handling.
	(3)	Understand and implement the concepts of Interfaces, Package and multithreaded programming.

	(4)	Design Graphical User Interface using applets and swing controls

DIGITAL CIRCUITS AND SYSTEMS

Course Code	**ECC 234**	**L-P-T-Cr.:**	**3**	**0**	**0**	**3**	**Semester:**	**III**

Category:	FC(OE)
Course Objectives:	To learn basics of digital electronic circuits and acquire knowledge on the logic gates, Boolean algebra, combinational and sequential circuits.

UNIT – I:		**(9 Hours)**

Review of Number System -binary, octal, decimal and hexadecimal number systems and conversions.1’s complements, 2’complement, binary addition, subtraction, multiplication & division. Logic gates and Boolean algebra: NAND & NOR Implementation, De Morgan’s law, Duality theorem, Digital Logic Gates for Multiple inputs. Boolean functions, Canonical & standard form; min terms & max term. The Map Method, K Map for two, three, four variables. Product of Sum (POS), Sum of product (SOP) simplification, Don’t care conditions. Error detection& correction: Parity Generator and Checker Circuit.

UNIT – II:		**(9 Hours)**

Combinational Logic Circuits and Logic Families: Analysis & Design of Binary Half Adder & Full Adder circuit, Carry Look Ahead adder. Half and Full-subtractor circuit, Decoders, Decoder for Seven segment display, decoder for binary to grey and grey to binary code. Encoders, Priority encoders, Multiplexers and Demultiplexers, Magnitude Comparator. MOS & C-MOS Logic circuits.,

UNIT – III:		**(9 Hours)**

Sequential Logic Circuit: Sequential Circuit, Latches, Flip-flop (S-R, J-K, D, T, M/S), edge triggering and level triggering.

Register &Counters: Universal Shift Register (SISO, SIPO, PISO, PIPO), Synchronous Counter, Ripple counter, Modulo-n Counter, Up-Down Counter, Asynchronous Counter, Analog to digital converter (ADC) & Digital to analog converters (DAC).

UNIT – IV:		**(9 Hours)**

Memory & Programmable Logic: Classification of memories–ROM, ROM organization, PROM, EPROM, EEPROM, EAPROM, RAM, RAM organization.

Programmable Logic Devices, Programmable Logic Array (PLA), Programmable Array Logic(PAL), Field Programmable Gate Arrays (FPGA).

TEXT BOOKS:

(1)	Digital Design, 3rd edition by M. Morris Mano, PHI

REFERENCE BOOKS:

(1)	Digital Fundamentals – Floyd & Jain, Pearson education
(2)	Digital Principles & Applications – Malvino, Leach & Saha, 6th Edition, Tata Mc Graw Hill
(3)	Switching Theory & Digital Electronics – V. K. Jain, Khanna Publishers

Course Outcomes:	After completion of this course the students should be able to:	
	(1)	Remember and understand the basic concepts/ Digital Circuit and System
	(2)	Analyze the various concepts to understand them through case studies
	(3)	Apply the knowledge in understanding practical problems
	(4)	Execute / Create the project or field assignment as per the knowledge gained in the course

ENGINEERING ECONOMICS AND COSTING

Course Code	**HSC 235**	**L-P-T-Cr.:**	**3**	**0**	**0**	**3**	**Semester:**	**III**
Category:	FC(HS)							

Course Objectives: The objective of this course is to acquaint the students with concepts and techniques in Economic Theory and to enable them to apply this knowledge in decision-making. Emphasis is given to changes in the nature of business firms in the globalization along with financial management idea and Cost management techniques in an organisation.

UNIT – I: **(10 Hours)**

Engineering Economics – Nature and scope, General concepts on micro & macroeconomics. The Theory of demand, Demand function, Law of demand and its exceptions, Elasticity of demand, Law of supply and elasticity of supply, Cost concept, Theory of production, Law of variable proportion, Law of returns to scale.

UNIT – II: **(10 Hours)**

Time value of money: Simple and compound interest, Time value equivalence, Compound interest factors, Cash flow diagrams, Calculation, Calculation of time –value equivalences. Present worth comparisons, Comparisons of assets with equal, unequal and infinite lives, comparison of deferred investments, Future worth comparison, payback period comparison.

Capital Budgeting Decision and Techniques.

UNIT – III: **(10 Hours)**

Analysis of public Projects: Benefit/ Cost analysis, quantification of project cost and benefits, benefit/ cost applications, Cost –effectiveness analysis.

Fixed and variable cost, Product and Process Costing, Standard Costing, Cost estimation, Relevant Cost for decision making, Cost estimation, Cost control and Cost reduction techniques

TEXT BOOKS:

(1) Horn green, C.T., Cost Accounting, Prentice Hall of India

(2) Riggs, J.L ., Dedworth, Bedworth, D.B, Randhawa, S.U. Engineering Economics, McGrawHill International Edition, 1996

(3) Financial Management, Van Horne, Prentice Hall

(4) Financial Management, Prasanna Chandra, Tata McGraw Hill

Course Outcomes: After completion of this course the students should be able to:

This course provides a thorough understanding about the economy and finance of the country along with the knowledge recent economic trends

COMPUTER ORGANIZATION AND ARCHITECTURE

Course Code	CSC 236	**L-P-T-Cr.:**	3	0	0	3	**Semester:**	**III**
Category:	PC(CS)							

Course Objectives:	1	How Computer Systems work & the basic principles
	2	Instruction Level Architecture and Instruction Execution
	3	The current state of art in memory system design
	4	How I/O devices are accessed and its principles and To provide the knowledge on Instruction Level Parallelism

UNIT – I: **(12 Hours)**

Introduction to Computer Architecture and Organization. Von Neuman Architecture, Flynn Classification. Register Transfer and Micro operations: Register transfer language, Arithmetic Micro-operations, Logic Micro-operations, Shift Micro-operations, Bus and memory transfers. Computer Organization and Design: Instruction cycle, computer registers, common bus system, computer instructions, addressing modes, design of a basic computer.

UNIT – II: **(12 Hours)**

Central Processing Unit: General register organization, stack organization, Instruction formats, Data transfer and manipulation, program control. RISC, CISC characteristics. Pipeline and Vector processing: Pipeline structure, speedup, efficiency, throughput and bottlenecks. Arithmetic pipeline and Instruction pipeline.

UNIT – III: **(12 Hours)**

Computer Arithmetic: Adder, Ripple carry Adder, carry look Ahead Adder, Multiplication: Add and Shift, Array multiplier and Booth Multiplier, Division: restoring and Non-restoring Techniques. Floating Point Arithmetic: Floating point representation, Add, Subtract, Multiplication, Division.

UNIT – IV: **(12 Hours)**

Memory Organization: RAM, ROM, Memory Hierarchy, Organization, Associative memory, Cache memory, and Virtual memory: Paging and Segmentation. Input-Output Organization: Input-Output Interface, Modes of Transfer, Priority Interrupt, DMA, IOP processor.

TEXT BOOKS:

(1)	Computer Organization and Architecture - William Stallings (Pearson Education Asia)
(2)	Computer Organization and Architecture -John P. Hayes (McGraw -Hill)

REFERENCE BOOKS:

(1)	Computer Organization -V. Carl. Hamacher (McGraw-Hill)
(2)	Computer Systems Architecture – M.Moris Mano, IIIrd Edition, Pearson/PHI
(3)	Computer Organization – Carl Hamacher, ZvonksVranesic, SafeaZaky, Vth Edition, McGraw Hill.
(4)	“Computer Architecture and Organization”, 3rd Edition by John P. Hayes,WCB/McGraw-Hill

Course Outcomes:	Upon completion of this course, students should be able to:	
	1	Student will learn the concepts of computer organization for several engineering applications.
	2	Student will develop the ability and confidence to use the fundamentals of computer organization as a tool in the engineering of digital systems
	3	An ability to identify, formulate, and solve hardware and software computer engineering problems using sound computer engineering principle
	4	To impart the knowledge on micro programming

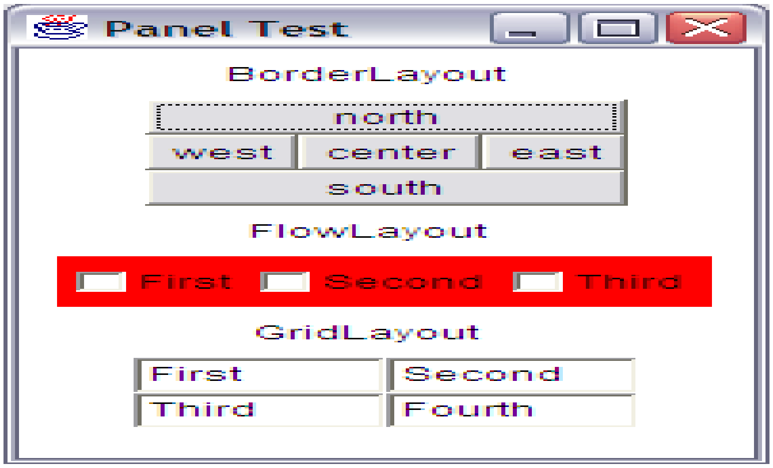
OBJECT ORIENTED PROGRAMMING USING JAVA LAB

Course Code	CSL 237	**L-P-T-Cr.:**	0	3	0	2	**Semester:**	**III**
Category:	PC(CS)							

Course Objectives:	(1)	Learn the concepts of object-oriented programming.
	(2)	Introduce the implementation of inheritance, packages and interfaces.
	(3)	Understand the concepts of exception handling and multithreading.
	(4)	Introduce the java collection framework and I/O classes.

List of Experiments

Lab-1	Write a java program to read three numeric values (integer) from user and find the largest number among them.
Lab-2	Write a program to print the Fibonacci series up to a given number taken from user through command line.
Lab-3	Write a statistical computation program that to find out the maximum, minimum and mean value. Read input through command line.
Lab-4	WAP to create a class Rectangle (length, breadth), with zero argument constructor (default value is 5.0), one argument constructor (length = breadth), and two argument constructors, and define the methods area and perimeter of the rectangle. Create different objects with the help of three different constructors and print the area (length x breadth) and perimeter (2 x (length + breadth)) of those objects.
Lab-5	Define a class called Room with the following attributes 1. length, 2. breadth, 3. height, 4. floor_area, 5. Wall_area, 6. No. of_fans, 7. No. of_windows, 8.no. of_doors. Define a suitable constructor and a method to display details of a room. Assume that 20% of the total wall area is occupied by doors and windows and calculate accordingly. All data must be taken from user.
Lab-6	Define a class point, inherit class line from point, rectangle from line, and cube from rectangle. Write no argument constructor in each class. Write a print statement in these constructors mentioning which class it is. Create an object of the cube class in the main method of a separate class called test and show the output.
Lab-7	WAP to create a Person class having name, age and gender as instance variables. Write three constructors for constructor overloading like, i. First with no-argument. ii. Second with three arguments for passing name, age and gender. iii. Third with object as parameter to create a new copy of an existing Person object. Display the properties of Person class object with suitable methods.
Lab-8	Create an abstract class Shape with methods calc_area and calc_volume. Derive four classes Sphere(radius) , Cone(radius, height) and Cylinder(radius, height), Box(length, breadth, height) from it. Calculate area and volume of all. (Use Method overriding).
Lab-9	Define an abstract class “Staff” with members name and address. Define two subclasses of this class – “FullTimeStaff” (department, salary) and “PartTimeStaff” (numberof-hours, rate-per-hour). Define appropriate constructors. Create n objects which could be of either FullTimeStaff or PartTimeStaff class by asking the user’s choice. Display details of all “FullTimeStaff” objects and all “PartTimeStaff” objects.
Lab-10	Define an interface “StackOperations” which declares methods for a static stack. Define a class “MyStack” which contains an array and top as data members and implements the above interface. Initialize the stack using a constructor. Write a menu driven program to perform operations on a stack object.
Lab-11	Define an interface “QueueOperations” which declares methods for a static queue.

	Define a class “MyQueue” which contains an array and front and rear as data members and implements the above interface. Initialize the queue using a constructor. Write a menu driven program to perform operations on a queue object.
Lab-12	Write a java program to create n objects of the Student class. Assign roll numbers in the ascending order using static method. Accept name and percentage from the user for each object. Define a method “sort Student” which sorts the array on the basis of percentage
Lab-13	Write a program to enter the student’s name, Rollno. Marks, in any no. of subjects as command line argument and find the percentage and grade of the student and thrown a NumberFormatException if required.
Lab-14	WAP having multiple catch and finally blocks where the catch blocks should handle the exceptions like, ArrayIndexOutOfBoundsException, NumberFormatException and ArithmeticException or any other exception.
Lab-15	Write a java program to creates ten threads, each of which do some work (search for the maximum value of a large matrix. Each thread searches one portion of the matrix.) It waits for them all to finish, then gathers the results.
Lab-16	Write a java program to show the use of synchronized method ().
Lab-17	Write a program to remove common characters from two strings.
Lab-18	Write a program to print all the palindrome words of a given string.
Lab-19	Input some strings through command line. Half of which will be stored in a String array and rest will be stored in a StringBuffer array. Write a program that will concatenate each element of this array of String objects with each element of StringBuffer objects. And the result will be stored in an array of StringBuffer.
Lab-20	Write an applet program to display the following by using different layouts.  Panel Test BorderLayout north west center east south FlowLayout First Second Third GridLayout First Second Third Fourth
Course Outcomes:	After completion of this course the students should be able to:
	(1) Understand the concepts and implement arrays and strings.
	(2) Understand the object-oriented programming concepts, solve real world problems and implement the concepts of exception Handling.
	(3) Understand and implement the concepts of Interfaces, Package and multithreaded programming.
	(4) Design Graphical User Interface using applets and swing controls

DIGITAL CIRCUITS LAB.

Course Code	ECL 238	**L-P-T-Cr.:**	0	3	0	2	**Semester:**	**III**
Category:	FC(OE)							
Course Objectives:	To verify the function of different logic gate ICs, truth tables and the working of different combinational and sequential logic circuits.							

List of Experiments

Lab-1	Verification of Logic Gates.
Lab-2	Realization of Gates Using NAND Gate.
Lab-3	Realization of Gates Using NOR Gate.
Lab-4	Realization of Half and Full Adder using Gates.
Lab-5	Realization of Encoder / Decoder (4:2 / 2:4).
Lab-6	Realization of Multiplexer / De-multiplexer (2:1 / 1:2).
Lab-7	Realization of Flip-Flop (RS, T, D, JK).
Lab-8	Realization of BCD to Seven Segment Display.
Lab-9	Realization of Shift Register (2-Bit).
Lab-10	Realization of Counters

Course Outcomes:	After completion of this course the students should be able to:	
	(1)	Remember and understand the basic concepts/ Principles of Digital Circuits
	(2)	Analyze the various concepts to understand them through case studies
	(3)	Apply the knowledge in understanding practical problems
	(4)	Execute / Create the project or field assignment as per the knowledge gained in the course