DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

CURRICULUM

FOR THE ACADEMIC YEAR 2021-22

III AND IV SEMESTER B.E.

POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING

(An autonomous college under VTU)

KALABURGI

About the Institution

Poojya Doddappa Appa College of Engineering was the first institution established by the Hyderabad Karnataka Education Society in 1958 by Sri. Mahadevappa Rampure, with an aim to impart quality education to the youth of Hyderabad Karnataka region. The college is spread over 71 acres of land and has sprawling complex housing student activity center, 1000 seat capacity auditorium and gardens and greenery around.

At present the total intake of the Institute stands at 980 in 11 UG programmers, 193 in10 PG programmers and 197 Research Scholars pursuing their Ph.D. in 12 Research centers. UGC has approved the academic Autonomy from 2009-10 to 2014-15 and extended up-to 2018-19. College has been declared fit under section 2(f) and 12 (B) of UGC Act1956. The Institute has been awarded with Bharatiya Vidya Bhavan National Award for an Engineering College having Best Overall Performance for the year 2017 by ISTE (Indian Society for Technical Education).

The College receives grant in aid funds from State Government for five departments. A number of research projects have been funded by MHRD/AICTE which includes also funds for modernization of laboratories. The institute is one among nine Institutions identified by Ministry of ITBT, Govt. of Karnataka for setting-up Incubation Centre. Innovative ideas of the students are encouraged to be translated into commercial products for the benefit of society. A fund of Rs.3 Lakh per project is being financed by Ministry of ITBT, Govt. of Karnataka. The institute is declared as the best NAIN Centre for submitting ten innovative projects for 2016. Several UG student projects are funded by Karnataka State Council for Science and Technology every year. The college has got a central library facility which houses also a separate digital library with 40PCs. The library has got large collection of books along with e-journals which is upgraded every year by allocation of necessary funds. Training & Placement Computer Centre with 120 Core i5 systems with LCD Screen has been setup to provide facility for conducting Training, Workshops, Seminars, Conferences, Online examination etc. Separate hostel facility is provided for boys and girls. An exclusive Examination Centre housing state of the art facilities has been constructed. 100Mbps internet facility and wi fi facility are provided for the students. The student activity center houses sports facilities, canteen facility and three seminar halls.

The college has acquired autonomous status for both UG and PG programs since 2007-08 and is one among six colleges in Karnataka State to acquire the autonomous status for both UG and PG programs. The college is one among 14 colleges selected under TEQIP scheme, sponsored by

World Bank. The college is selected for TEQIP-III as Mentoring Institution for BIET, Jhansi, UP by NPIU New Delhi.

The Institute, taking the advantage of academic autonomy, believes in preparing the students through OBE adopting Choice Based Credit System (CBCS) and Continuous Assessment and Grading Pattern of examination. Also a one year diploma course in Interior design has been started by the Architecture department. The curriculum is framed with wide interaction with alumni and Industry resource persons so as to impart necessary updated skills in the students.

Highly qualified and dedicated human resource, state of the art facilities, effective institute- alumni and industry-institute interaction make the learning process a unique and satisfying experience at Poojya Doddappa Appa College of Engineering.

About the Department

The Information Science & Engineering Department is established in the year 2000 with an intake of 40. The present intake of department is 60 with total of 180 students studying in I, II, III and IV year. Department is recognized as Research Centre by Visvesvaraya Technological University Belagavi in the year 2013 and at present 14 research scholars are pursuing their Ph.D., Research centre put forward its activities and grabbed the AICTE research promotion schemes to carry out research project.

The faculty and staff of the department made remarkable standings by involving themselves in academic activities like conducting various Workshops, Seminars and Industry Institute Interaction programs. The workshops and seminars are sponsored by TEQIP- I & II, ISTE, IE and IETE.

Department has signed MoU with various industries like Global Edge Technologies, Mind solvit Pvt. Ltd., Clusters Info Solutions, Nihon Communications, Crimson Innovative Technologies KGTTI, EDUWIZE and Red Hat Academy, that works like bench mark for the students and faculty to get expose to IT sector and current industry trends.

Vision of the Institution

To be an institute of excellence in technical education and research to serve the needs of the industry and society at local and global levels

Mission of the Institution

- To provide a high-quality educational experience for students with values and ethics that enables them to become leaders in their chosen profession.
- To create, explore, and develop innovations in engineering and science through research and development activities.
- To provide beneficial service to the local, state, national and international industries and communities via educational, technical, and professional activities.

Department Vision

To impart quality education and research in Information Technology to produce a competent, committed and goal oriented workforce to fulfill the needs of the local and global requirements.

Department Mission

- To produce quality workforce with cutting edge technology.
- To engage in innovative teaching learning, research and community service.
- To strengthen continuous interactions with alumni and industry and enable the graduates to attain entrepreneur qualities with life-long learning skill.

Program Educational Objectives

PEO 1: Apply the principles of information and allied science, mathematics and scientific investigation to solve real world problems appropriate to thediscipline.

PEO 2: Apply current industry accepted computing practices and emerging technologies to analyze, design, implement, and verify high quality IT-based solutions to real world problems.

PEO 3: Exhibit teamwork and effective communication skills**.**

PEO 4: Understand the ethical obligations, social impacts and apply their technical knowledge positively and appropriately in the course of career and professional journey**.**

PEO 5: Be successfully employed or accepted into a postgraduate program, and demonstrate a pursuit of life long learning**.**

Program Outcomes

Engineering Graduates will be able to:

Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

Design/development of solutions: Design solutions for complex 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.

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.

Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

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 engineering practice.

Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multi disciplinary settings.

Communication: Communicate effectively on complex engineering activities with the 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.

Project management and finance: Demonstrate knowledge and understanding of the 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 multi disciplinary environments.

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 technological change.

Program Specific Outcomes (PSOs)

The graduates are able to:

PSO1: Demonstrate the working principles of computing systems and application Software

PSO2: Provide the suitable strategies and design solutions to solve IT related problems.

PSO3: Apply the professional practices to implement and deploy the real life applications

POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING, KALABURAGI
Choice Based Credit System (CBCS)
Scheme of Teaching and Examination 2019 – 20
(Effective from the academic year 2019-20)

III Semester

Sl. No.	Course and Course Code		Course Title	Teaching Department	Teaching Hours/Week			Examination				Credits
					Theory Lecture	Tutorial	Practical/ Drawing	Duration in hours	SEE Mark	CIE Mark	Total Marks	
1.	BS	19MA31	Computational Methods for Information Science	Mathematics	3	--	--	03	50	50	100	3
2.	ES	19IS32	Discrete Mathematics and Graph Theory	ISE	3	--	-	03	50	50	100	3
3.	PC	19IS33	Logic Design	ISE	3	--	--	03	50	50	100	3
4.	PC	19IS34	Data Structures Using C	ISE	3	--	--	03	50	50	100	3
5.	PC	19IS35	Computer Organization	ISE	3	--	--	03	50	50	100	3
6.	PC	19IS36	Object Oriented Programming with C++	ISE	4	--	--	03	50	50	100	4
7.	NCMC	19HU36	Constitution of India and Professional Ethics	Humanities/ Civil Dept	2	--	--	02	50	50	100	0
8.	PC	19ISL31	Logic Design Lab	ISE	--	--	2	03	50	50	100	1
9.	PC	19ISL32	Data Structure Lab	ISE	--	-	2	03	50	50	100	1
10.	PC	19ISL33	OOPS Lab	ISE	--	--	2	03	50	50	100	1
Tota l					**21**	**--**	**06**	**29**	**500**	**500**	**1000**	**22**
Note: Auto, IPE, CCT, Mechanical, Civil – Kannada of 1 credit course, CIV EE,EI,ISE,CSE,ECE – CIP												
Course prescribed to lateral entry Diploma holders admitted to III semester of Engineering programs												
11	NCM C	19MAD031	Additional Mathematics-I	Mathematics Dept.	03	--		03	50	50	100	0

POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING, KALABURAGI
Choice Based Credit System (CBCS)
Scheme of Teaching and Examination 2019– 20
(Effective from the academic year 2019 – 20)

IV Semester

Sl. No.	Course and Course Code		Course Title	Teaching Department	Teaching Hours/Week			Examination				Credits
					Theory Lectur	Tutorial	Practical/ D	Duratio n in	SEE	CIE Marks	Total Marks	
1.	BS	19MA41	Mathematical Methods For Information Science	Mathematics	4	--	--	03	50	50	100	4
2.	ES	19IS42	Python Programming with UNIX System	ISE	4	--	--	03	50	50	100	4
3.	PC	19IS43	Finite Automata and Formal Languages	ISE	3	--	--	03	50	50	100	3
4.	PC	19IS44	Analysis and Design of Algorithm	ISE	3	--	--	03	50	50	100	3
5.	PC	19IS45	Microprocessor and ARM Processor	ISE	4	-	--	03	50	50	100	4
6.	NCMC	19CV46	Environmental studies	Civil Dept	2	--	-	02	50	50	100	0
7.	KAN	19KAN47	Kannada	Humanities	1	--	--	01	-	50	50	1
8.	PC	19ISL41	Python and UNIX Shell Programming Lab	ISE	-	--	2	03	50	50	100	1
9.	PC	19ISL42	Algorithms Lab	ISE	--	--	2	03	50	50	100	1
10.	PC	19ISL43	Microprocessor and ARM Processor Lab	ISE	--	--	2	03	50	50	100	1
Total					**21**	**--**	**06**	**27**	**450**	**500**	**950**	**22**

Note: Auto, IPE, CCT, Mechanical, Civil – CIP EE,EI,ISE,CSE,ECE – Kannada of 1 credit course, CIV

Course prescribed to lateral entry Diploma holders admitted to III semester of Engineering programs

11	NCMC	19MAD041	Additional Mathematics-II	Mathematics Dept.	03	--		03	50	50	100	0

PDA COLLEGE OF ENGINEERING,KALABURAGI

B E Third Semester

COMPUTATIONAL METHODS FOR INFORMATION SCIENCE

Subjectcode:19MA31		**Credit: 03**
CIE:50 Marks	**SEE:50Marks**	**SEE: 3 hours**
Hours/Week:3Hrs. (Theory)		**Total hours:28**

Prerequisite: Students should have knowledge of Differential calculus, Integral calculus and Differential equations.

Course objectives: To enable the students to obtain the knowledge of Engineering Mathematics in the following topics

1. **Numerical methods to solve algebraic and Transcendental equations**
2 **Interpolation methods and Numerical integration**
3 **Numerical solution of ODE and PDE**
4. **Methods of least squares to fit straight line and second degree parabola**
5. **Solve the problems using probability theory.**

Module-I 6hours

Errors And Approximations:
Errors in arithmetic operations, Errors in function, approximation by Taylor's series.

Algebraic And Transcendental Equations: Solution by Bisection and Newton's Raphson methods.
Finite differences: (Forward and Backward differences), Interpolation, Newton's Forward and Backward formulae. Lagrange's interpolation formula .

Module-II 5hours

Numerical differentiation and Integration Numerical differentiation using Newton's forward and backward interpolation formulae and problems..

Numerical integration: Trapezoidal rule, Simpsons 1/3rd and 3/8th rule, Weddle's rule (all formulae and rules without proof).

Module -III

Numerical solution of ODE and PDE **6 hours**

Numerical solution of Simultaneous ordinary differential equations of first order and second order differential equations by Runge-kutta method, Numerical solution of one dimensional heat and wave equation by explicit method Laplace equation using standard five point formula

Module -IV

Statistical methods: **6hours**

Curve fitting by the method of least squares:
$y = ax + b,\quad y = a + bx + cx^2,\quad y = ab^x$ Correlation, Rank correlation, lines of regression and angle between two regression lines.

Module-V

Introduction to Probability: **5 hours**

Probability: Addition rule and multiplication rule conditional probability Bay's rule with examples .

COURSE OUTCOMES:

On completion of this course, students are able to:

CO1: Solve numerically the algebraic and transcendental equations and Compute polynomials using interpolating techniques.

CO2: Compute derivatives of the functions numerically using given data.

CO3: Apply numerical techniques to solve ordinary and partial differential equations.

CO4: Apply the method of least square to estimate the parameters in regression model.

CO5: Solve problems using probability rules.

Text book:

Higher Engineering Mathematics by B.S.Grewal, Khanna publishers; 40th Edition.2007

Reference books:

1.Advanced Engineering Mathematics by E. Kreyszig, John Willey & sons 8thEdn.

2.Advanced Engineering Mathematics by R.K.Jain& S.R.K Iyengar; Narosa publishing House.

3.Introductory methods of numerical analysis by S.S.Sastry

DISCRETE MATHEMATICS AND GRAPH THEORY		
Subject Code	**19IS32**	Credits:03
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 03 Hours (Theory)		Total hours: 42

Prerequisite: The Students must be familiar with basic arithmetic and algebraic operations.

Course Objectives:

To enable the students to obtain the knowledge of Discrete Mathematics And Graph Theory in the following topics.

- Understand computational aspects of sets, relations and mathematical logic.
- Understand various types of functions and operations on functions.
- Understand various types of graphs.
- Determine a connectivity of graph using the concepts of graph theory.
- Understand concept of trees and various methods of the tree traversals.

Modules	**Teaching Hours**
Module-I **Set Theory:** Sets and Subsets, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams, A First Word on Probability, Countable and Uncountable Sets. Fundamentals of Logic: Basic Connectives and Truth Tables, Logic Equivalence The Laws of Logic, Logical Implication – Rules of Inference	**9 Hours**

Module-II **Fundamentals of Logic contd.:** The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems, Mathematical Induction, **Relations and Functions:** Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions – Stirling Numbers of the Second Kind, Special Functions, The Pigeon-hole Principle.	**8 Hours**
Module-III **Relations and Function contd.:** Function composition and inverse functions, Properties of Relations, Computer Recognition – ZeroOne Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions.	**8 Hours**
Module-IV **Introduction to Graph Theory:** Definitions and Example, Sub graph, Complements and Graph Isomorphism, Vertex Degree, Euler Trails and Circuits. Planar graphs, Hamilton Paths and Cycles, Graph coloring, Chromatic Polynomials, Graph colouring, chromatic Polynomials;	**9 Hours**
Module-V **Trees:** Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes Connectivity: Cut-Vertices, Blocks, Connectivity, Menger's Theorem, Exploration: Geodetic Sets. **Traversability:** Eulerian Graphs, Hamiltonian Graphs, Exploration: Hamiltonian Walks and Numbers, Excursion: The Early Books of Graph Theory.	**8 Hours**

Question paper pattern:

1. The question paper will have TEN questions.
2. There will be TWO questions in each module, covering all the topics.
3. The student need to answer FIVE full questions, selecting ONE full question from each module.

Text books:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education.2014.
2. "Introduction To Graph Theory" By Gary Chartrand ,Ping ZhangTMH,2006

Reference Books:

1. C. L. Liu C. L., “Elements of Discrete Mathematics”, 2nd Edition, McGraw Hill,Singapore,
2. J.P. Tremblay, “Discrete Mathematical Structures with Applications to Computer Science”, McGraw Hill,N.Y.,
3. Kenneth H Rosen, “Discrete Mathematics and its applications”, 6th Edition, McGrawHill
4. Discrete Mathematical structures, Dr. D. S. Chandrashekariah.Prism,

5. B.Kolman and R.C.Busby, “Discrete Mathematical Structures for Computer Science”, PHI, NewDelhi,

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Understand the basic principles of set theory
	CO2	Apply fundamentals of logic in solving problems
	CO3	Make use of functional, relational and Boolean operations in solving problems
	CO4	Describe and solve problems using concepts of graph theory
	CO5	Analyze various types of trees and tree traversing techniques

LOGIC DESIGN

Subject Code	**19IS33**	Credits:03
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 03 Hours (Theory)		Total Hours: 42

Prerequisite: The students should have the understanding of basic electronics, basic electrical engineering and algebraic laws.

Course Objectives: To enable the students to obtain the knowledge of Logic Design in the following topics.

- Introduce the basics of Minimizing Booleans functions by using various techniques like K-Map and Quine Mclusky methods and implement by using suitable Logic gates.
- Discuss the combinational logic circuits like Full Adder, Subtractor, Magnitude Comparators, Code Converters etc. and implement by using logic gates/ICs.
- Present the working of various Flip- Flops, Register types, Counters, ADC/DAC and their applications.

Modules	**Teaching Hours**
Module I **The Basic Gates**: Review of Basic Logic gates, Positive and Negative Logic. **Combinational Logic Circuits:** 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 simplifications, Simplification by Quine-McClusky Method, Hazards and Hazard covers.	**8 Hours**
Module II **Data-Processing Circuits**: Multiplexers, Demultiplexers,1-of-16 Decoder, BCD to Decimal Decoders, Seven Segment Decoders, Encoders, Exclusive-OR Gates, Parity Generators and Checkers, Magnitude Comparator, Programmable Array Logic, Programmable Logic , Arithmetic Building Blocks, Arithmetic Logic Unit, Clocks, Clock Waveforms, TTL Clock, Schmitt Trigger.	**9 Hours**

Module III **Flip Flops:** RS Flip-Flops, Gated Flip-Flops, Edge-triggered RS FLIP- FLOP. Clocked D FLIP-FLOP, Edge-triggered D FLIP-FLOP, Edge- triggered JK FLIP-FLOP, FLIP-FLOP Timing, JK Master-slave FLIP- FLOP, Switch Contact Bounce Circuits, Various Representation of FLIP- FLOPs, Analysis of Sequential Circuits.	**8 Hours**
Module IV **Registers:** Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers. **Counters:** Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus. Decade Counters, Pre settable Counters, Counter Design as a Synthesis problem, A Digital Clock.	**9 Hours**
Module V **D/A Conversion and A/D Conversion:** Variable, Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution, A/D Converter Simultaneous Conversion, A/D Converter-Counter Method, Continuous A/D Conversion, A/D Techniques, Dual-slope A/D Conversion, A/D Accuracy and Resolution.	**8 Hours**

Question paper pattern:

3. The question paper will have TEN questions.
4. There will be TWO questions in each module, covering all the topics.
5. The student need to answer FIVE full questions, selecting ONE full question from each module.

Text books:

1. Donald P Leach, Albert Paul Malvino& Goutam Saha: Digital Principles and Applications, 8th Edition, Tata McGraw Hill, 2015.

Reference Books:

1. Stephen Brown, ZvonkoVranesic: Fundamentals of Digital Logic Design with VHDL, 2nd Edition, Tata McGraw Hill,2012.
2. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson,2010.
3. Charles H. Roth: Fundamentals of Logic Design, Jr., 5th Edition, Cengage Learning,2004.
4. Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss: Digital Systems Principles and Applications, 10th Edition, Pearson Education,2007.
5. M Morris Mano: Digital Logic and Computer Design, 10 th Edition, Pearson Education, 2008

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	CO #	Course Outcome (CO)
	CO1	Discuss the concepts of basic gate and construct Logic Circuits using different simplification methods.
	CO2	Analyze and Design data processing circuits using various logical blocks.
	CO3	Analyze various types of Flip Flops using sequential circuits.
	CO4	Design various synchronous and asynchronous counters using registers and flip flops.
	CO5	Apply the analog and digital conversion techniques for the given data to design A/D and D/A conversion circuit.

DATA STRUCTURES USING C		
Subject Code	**19IS34**	Credits: 03
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 03 Hours (Theory)		Total Hours: 42

Prerequisite: The Students should have the thorough knowledge of C fundamentals.

Course Objectives:

To enable the students to obtain the knowledge of Data Structures With C in the following topics.

- Understand the concepts of data structures and algorithms.
- Understand the basic principles of dynamic memory allocation
- Understand the different data structures like stacks, queues, lists and trees
- Understand the search and sort techniques.

Modules	Teaching Hours
Module I **Pointers:** Pointers and Dynamic memory allocation, Data Abstraction, Arrays and Structures, Dynamically Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices.	**8 Hours**

Module II **Stacks:** Stacks Using dynamic Arrays, Evaluation of Expression: Expressions, Evaluating Postfix Expressions, Infix to Postfix, Recursion.	**8 Hours**
Module III **Queues:** Linear Queues, Circular queues using Dynamic Arrays, A Mazing Problem. Linked Lists: Singly Linked lists and Chains, Representing Chains in C, Linked Stacks and Queues, Polynomials.	**8 Hours**
Module IV **Linked List:** Additional List operations, Circular Linked Lists, Doubly linked Lists. Trees: Introduction, Binary Trees, Binary Tree Traversals.	**9 Hours**
Module V **Trees:** Additional Binary Tree Operations, Threaded Binary Trees. Binary Search Trees, Sorting: Insertion sort, Quick sort, Merge sort, Heap sort, Hashing- static and Dynamic Hashing .	**9 Hours**

Question paper pattern:

1. The question paper will have TEN questions.
2. There will be TWO questions in each module, covering all the topics.
3. The student need to answer FIVE full questions, selecting ONE full question from each module.

Text books:

1. Fundamentals of Data Structures in C, by Horowitz, Sahni, Anderson-Freed, 2nd Edition, Universities Press,2018.

2. Data Structures Using C and C++, by Yedidyah, Augenstein, Tannenbaum, 2nd Edition, Pearson Education,2003.

Reference Books:

1. Classic Data Structures by Debasis Samantha, 2nd Edition, PHI,2009.

2. Data Structures: A Pseudocode Approach with C by Richard F. Gilberg and Behrouz A. Forouzan, Cengage Learning,2005.

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Demonstrate the principles of dynamic memory allocation and basic data structures such as arrays, structures and unions.
	CO2	Analyze and implement different stack operations and recursive algorithms
	CO3	Analyze and implement queue and linked list operations to solve problems
	CO4	Implement tree traversal techniques using C Programming
	CO5	Analyze and Implement Algorithm for solving problems like sorting, searching, insertion and deletion of data

COMPUTER ORGANIZATION

Subject Code	**19IS35**	Credtis:03
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 03 Hours		Total Hours: 42

Prerequisite: The students should have basic knowledge of different units of computer.

Course Objectives : The students are able

1. Understand how-to measure the performance of computer and to execute the instructions.

2. Design how the Study the basic organization and architecture of digital computers (CPU, memory, I/O, software).

3. To perform how the mathematical operations are performed using number representation systems.

4. Understand the working of hardwired control and micro program control systems to generate control signals.

Modules	**Teaching Hours**
Module I **Basic Structure of Computers**: Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. **Machine Instructions and Programs:** Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines.	**9 Hours**

Module II **Input/Output Organization**: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB	**8 Hours**
Module III **Memory System**: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations, Virtual Memories And Secondary Storage.	**8 Hours**
Module IV **Arithmetic**: Numbers, Arithmetic Operations and Characters, 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.	**8 Hours**
Module V **Basic Processing Unit**: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro programmed Control. Basic Concepts of pipelining, Examples of Embedded Systems, Processor chips for embedded applications.	**9 Hours**
Question paper pattern: 1. The question paper will have TEN questions. 2. There will be TWO questions in each module, covering all the topics. 3. The student need to answer FIVE full questions, selecting ONE full question from each module.	
Text books: 1. Carl Hamacher, ZvonkoVranesic, SafwatZaky: Computer Organization, 5th Edition, Tata McGraw Hill, 2011.	
Reference Books:1. William Stallings: Computer Organization & Architecture, 7th Edition, PHI, 2006	
Course outcomes: **On completion of the course, the student will have the ability to:**	

Course Code	CO #	Course Outcome (CO)
	CO1	Apply instruction sequencing to develop assembly language programs and implement stacks, queues and subroutines.
	CO2	Analyze various methods to handle interrupts and memory accessing methods.
	CO3	Analyze different storage devices, memory management and virtual memory concepts.
	CO4	Illustrate arithmetic operations using different algorithms.
	CO5	Synthesize control signals using hardwired and micro program control technique.

OBJECT ORIENTED PROGRAMMING WITH C++

Subject Code	**19IS36**	Credits:04
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 04 Hours		Total Hours: 52

Prerequisite: The Student should have the thorough knowledge of C programming principles and Structures.

Course Objectives:

To enable the students to obtain the knowledge of Object Oriented Programming With C++ in the following topics.

- Understand the concepts object oriented programming paradigm.
- Understand the OOP features like Inheritance, Virtual Functions and Dynamic Polymorphism.
- Understand stream handling mechanism and operator overloading.
- Understand and analyze exception handling mechanism.

Modules	Teaching Hours

Module I **Introduction to C++:** A Review of Structures, Procedure-Oriented Programming Systems, Objected Oriented Programming Systems Comparison of C++ with C, Console Input/ Output in C++, Variables in C+ Reference Variables in C++, Function Prototyping, Function Overloading Default Values for Formal Arguments of Functions, Friend Functions, Inline Functions. **Class and Objects:** Introduction to Classes and Objects.	**10 Hours**
Module II **Class and Objects** contd.*:* Member Functions and Member Data, Objects and Functions, Objects and Arrays, Namespaces, Nested Classes. **Dynamic Memory Management:** Introduction, Dynamic Memory Allocation, Dynamic Memory Deal location, the set_new_handler () function **Constructors and Destructors:** Constructors, Constructors overloading, Destructors	**11 Hours**
Module III **Inheritance:** Introduction to Inheritance, Base Class and Derived class Pointers, Function Overriding, Base Class Initialization. The Protected Acces Specifier, Deriving by Different Access Specifiers, Different kinds of inheritance, Order of invocation of Constructors and Destructors Virtual functions and dynamic polymorphism : the need for virtual functions, virtual functions, the mechanism of virtual functions, pure virtual functions	**10 Hours**
Module IV **Stream handling**: Streams, The class hierarchy of handling streams, text and binary, input/output, text versus binary files, text input/output, binary input/output, opening and closing files, files as objects of the fstream class, File pointers, random access to files, error handling. **Operator overloading:** operator overloading, overloading the various operators-overloading the increment and the decrement operators (Prefix and postfix), overloading the unary minus and the unary plus operator, overloading the arithmetic operators.	**10 Hours**

Module V	
Operator overloading contd.: Overloading the relational operators, overloading the assignment operator, overloading the insertion and extraction operators, overloading the new and the delete operators, overloading the subscript operator, **Templates:** Introduction, function templates, class templates. **Exception handling:** Introduction, C-style handling of error generating codes, C++ style solution – the try/throw/catch construct. Limitation of exception handling.	**11 Hours**

Question paper pattern:

1. The question paper will have TEN questions.
2. There will be TWO questions in each module, covering all the topics.
3. The student need to answer FIVE full questions, selecting ONE full question from each Module

Text books:

1. Object Programming With C++, Sourav Sahay, Oxford University Press,2006. (Chapter 1-10)

Reference Books:

1. C++Primer, StanelyB.Lipman, JoseeLajoie,Barbara E.Moo,4th Edition , Addision Wesely,2012.
2. The Complete Reference C++,Herbert ,4thEdition,TMH,2017

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	CO #	Course Outcome (CO)
	CO1	Analyze the Principles of Object Oriented Programming Paradigm.
	CO2	Implement Class and objects using C++ programming techniques and apply data encapsulation.
	CO3	Implement Object Oriented Program features like inheritance and dynamic polymorphism.
	CO4	Analyze the importance of stream handling and random access of files.
	CO5	Analyze the different operator overloading and Exception Handling techniques using C++.

CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS

Subject Code	**19HU36**	Credits:00
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 02 Hours		Total Hours: 28

Prerequisite:

Course Objectives:

To enable the students to obtain the basic knowledge about The Constitution of India and Professional Ethics in the following topics:-

- Introduction and Fundamental Rights
- Directive Principles of the State Policy and the State Executive
  The Union Executive
- Constitutional Provisions for women, Children & SC/ST ‘S , Emergency Provisions and Election Process
- Engineering Ethics

Modules	Teaching Hours
Module I **Introduction and Fundamental Rights** : The Constitution of India. Evolution of the Constitution. The Constituent Assembly of India. Sources and Features of the Indian Constitution. Preamble to the Constitution of India. Salient Features of Fundamental Rights and their classification. General exercise of Fundamental Rights and their limitations. RTI (Right to Information Act of 2005 Under Article 19(1)) and The Right of Children to Free and Compulsory Education Act or Right to Education Act (RTE) Under Article 21-A of the Constitution. Article 371(J) of the Constitution applicable to Hyderabad Karnataka Area.	**6 Hours**
Module II **Directive Principles of the State Policy and The State Executive**: Under Article 36 to 51 of The Constitution and their Relevance. Fundamental Duties Under Article 51A of The Constitution and their Relevance. Stat Government - The Governor- Appointment, Powers and Functions of th Governor. The Appointment of Chief Minster, his Powers and Functions The State Council of Ministers and their Functions. The State legislature and The State Council. The High Court of the State, its Powers an Jurisdiction. Appointment and Qualifications of High Court Judges.	**6 Hours**
Module III **The Union Executive**: Central Government. The President of India, hi lection, Powers and Functions. The Vice-President of India, his Election powers and Functions. The Supreme Court of India and its Structure Appointment and Qualification of Supreme Court Judges. Their Powers unction's. The Structure of Judiciary in India. The Parliament of India. The Prime Minister his Appointment, Powers and Functions. The Union Council of Ministers their Powers and Responsibilities. Concept of Public Interest litigation (PIL)	**6 Hours**
Module IV **Constitutional Provisions and Emergency Provisions and Election Process:** Constitutional for Women, Children, Backward Classes and scheduled Caste and Scheduled Tribes under different Article of The Constitution. Different types of Emergencies under Article 352, 356 and360 f the Constitution of India. The Election Commission of India- its Powers and Functions. The State Election Commission	**6 Hours**

Module V **Engineering Ethics :** Its Aims and Scope, Responsibilities of Engineers, impediments to their Responsibilities, Honesty, Integrity, Reliability, Risk and Safety Measures, Liabilities of Engineers	**4 Hours**

Question paper pattern:

1. The question paper will have TEN questions.
2. There will be TWO questions in each module, covering all the topics.
3.The student need to answer FIVE full questions, selecting ONE full question from each Module
4.

Text books:

1. An introduction to the constitution of India and Profession Ethics. By B. R. Venkatesh and Merunandan K. B. Publisher : Idea International Publication Bangalore.
2. The Constitution of India and Professional Ethics. By K. R. Phaneesh. Publisher : Sudha Publication Bangalore.
3. Professional Ethics. By S. Chand. Publisher : S. Chand & Company Ltd. Ram Nagar, New Delhi -110055.

Reference Books:

1. Constitutionof India and Professional Ethics By : M Raja Ram. Publisher : New Age International(P) Limited, NewDelhi.
2. The Constitution allow of India By : J.N. Pandhey . Publisher : Central Law agency , Allahabad.

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Explain the evolution and features of constitution, fundamental rights and their classification
	CO2	Describe the directive principles of state policy, fundamental duties and The State Executive.
	CO3	Describe about The Union Executive and concept of Public Interest Litigation.
	CO4	Explain the Constitutional Provisions for women, children, SC/ST'S, Emergency Provisions and Election Process.
	CO5	Identifies the qualities required for an professional engineers to be ethical.

LOGIC DESIGN LAB		
Subject Code	**19ISL31**	Credits:01
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week: 02 Hours (Practical)		Total Hours:14

Prerequisite: The students are expected to have Knowledge of Basic Electronics and basic concepts in logic design.

Course Objectives: To enable the students to obtain the knowledge of Logic Design Lab in the following topics.

- Design basic logic circuits and analyze the operation of combinational circuits like the decoder, multiplexer, full adder.
- Analyze the operation of a flip-flop, counters and shift registers.
- Perform and interpret parameters such as voltage and time period using oscilloscopes.
- Design and analyze sequential logic circuits.

Experiments

1. Design and implement Half adder, Full Adder, Half Subtractor, Full Subtractor using basic gates.

2. Given any 4-variable logic expression, simplify using Entered Variable Map and realize the simplified logic using 8:1 multiplexer IC.
3. Realize a full adder using 3-to-8 decoder IC and 4 input NAND gates.
4. Design and implement code converter I) Binary to Gray II) Gray to Binary Code using basic gates.

5. Design Schmitt trigger circuits using Op-Amp for given UTP and LTP values.

6. Realize a J-K Master/Slave Flip-Flop using NAND gates and verify its truth table.
7. Design and implement a mod-n(n<8) synchronous up counter using J-K Flip-Flop Ics.

8. Design and implement a ring counter using 4-bit shift register.

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

10. Design a 4-bit R-2R ladder D/A converter using Op-Amp. Determine its accuracy and resolution.

Question paper pattern:

In SEE, students will be asked to execute one program which may be related to the above list of programs.

Reference: Lab Manual

Course outcomes:
On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Design and evaluate logical circuits using k-map and Map Entered Variable concepts
	CO2	Design and implement sequential circuits.
	CO3	Design and Implement counters & shift registers
	CO4	Design and evaluate the code converter using op-amp circuits.
	CO5	Design and evaluate timing and multi vibrator circuits.

DATA STRUCTURES LAB

Subject Code	**19ISL32**	Credits:01
CIE : 50	SEE : 50	SEE : 03 hrs
Hours / Week : 02 Hours (Practical)		Total Hours: 14

Prerequisite: The Students should have a basic knowledge of C programming language.

Course Objectives:
To enable the students to obtain the knowledge of Data Structures Lab in the following topics.

- Build capability to analyze and understand the program.
- Recognize the problem properties where stacks and queues are appropriate data structures.
- Apply different operations of stacks, queues and linked lists.
- Understand and apply various tree traversal methods.
- Apply the solutions for real life problems.

Experiments

1. Write a C program to create a sequential file with at least five records. Each record having the structure shown below.

	Name	Marks1	Marks2	Marks3
Non-Zero positive integer	25-characters	Positive integer	Positive integer	Positive integer

Write necessary function
a) To display all the records in the file
b) To search for a specific record based on the USN . In case the record is not found. Suitable message should be displayed. Both the options in this case must be demonstrated.

2. Write a C program, which accepts the Internet protocol(IP)address in decimal dot format(ex 153.18.8.105) and converts it into 32-bit long integer(ex.2568095849) using start library function and unions.

3. Write a program using Recursion i) Solving the Towers Of Hanoi problem ii) Searching an element on a given list of integers using the Binary search method.

4. Write a program to perform push, pop and display operations on a stack using
i) Linear array ii)pointers.
The program should print appropriate messages for stack overflow, stack underflow and stack empty.

5. Write a C program to convert and print a given valid parenthesized infix arithmetic expression to a postfix expression. The expression consists of single character operands and the binary operators +(plus),-

(minus),*(multiply) and /(divide).

6. 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 +(plus),-(minus),*(multiply) and/(divide).

7. Write a program to perform insert, delete and display operations on a queue using
i) Linear array ii)pointers.
The program should print appropriate messages for queue full and queue empty.

8. Write a C program to simulate the working of a circular queue of integers using an array. Provide the following operations

a)Insert
b)Delete
c)Display

9. Write a C program using dynamic variables and pointers, to construct a singly linked list consisting of the following information in each node: student id(integer),student name(character string) and semester(integer).The operations to be supposed are:
a) The insertion operation
i) At the front of a list
ii) At the back of the list
iii) At any position in the list
b) Deleting a node based on student id. If the specified node is not present in the list an error message should be displayed. Both the operations should be demonstrated.
c) Searching a node based on student id and update the information content. If the specified node is not present in the list an error message should be displayed. Both situations should be displayed.
d) Displaying all the nodes in the list.

10. 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 thelist.

11. Write a C program using dynamic variables and pointers to construct a stack of integers using singly linked list and to perform the following operations.

a) PUSH

b) POP

c) Display

The program should print appropriate messages for stack overflow, and stack empty.

12. Write a C program using dynamic variables and pointers to construct a queue of integers using singly linked list and to perform the following operations.

a) Insert

b) Delete

c) Display

The program should print appropriate messages for queue full and queue empty.

13. Write a C program

a) To construct a binary search tree of integers.

b) To traverse the tree using all the methods i.e, in order, preorder and post order

c) To display the elements in the tree.

14. Write a C program to evaluate an expression tree.

Question paper pattern:

In SEE, students will be asked to execute the program which may be related to the above list of programs

Reference: Lab Manual

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	CO #	Course Outcome (CO)

	CO1	Implement simple data structure using C Program
	CO2	Implement stacks, queues and linked lists
	CO3	Implement array and pointer operations on stacks, queues and linked lists
	CO4	Implement different sorting and searching techniques using C
	CO5	Apply the different data structures for implementing solutions to practical problems

OBJECT ORIENTED PROGRAMMING LAB

Subject Code	**19ISL33**	Credits:01
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 02 Hours (Practical)		Total Hours: 14

Prerequisite: The Students should have the knowledge of programming in C and C++ and problem solving and debugging methodologies.

Course Objectives:
To enable the students to obtain the knowledge of OBJECT ORIENTED PROGRAMMING USING C++ Lab in the following topics.

- Familiarize with object oriented concept and their implementation in C++.
- Facilitate with skills required to solve problems using object oriented concepts.
- Impart the knowledge required to write a code with good programming practices.
- To apply the event and exceptions handling mechanism.

Experiments

1) Create a simple class **STUDENT** containing data members roll no, name age & display the contents using setdat() and Outdata()methods.

Test the program with

a) Member function inside the body of the student class.

b) Member function outside the body of the student class (using ::operator).

2) Write a C++ program to create class **DATE** and member function day, month, year. Display age of the person by considering date of birth and current date using inline function.

3) Write a C++ program to create a class **ACC** with data members, accno, balance. Create objects ACC1, ACC2 andACC3. Write a member function to transfer money from ACC3 to ACC1.Display the balance in all accounts.

4) Create a class called **QUEUE** perform insertion and deletion of elements from the queue using constructors and destructors.

5) Write a C++ program to sort N numbers using swap as friend function.

i) Write a C++ program to create a class called **STACK** using an array of integers. Implement the following operations by overloading + &-.s1=s1+ element; where s1 is an object of the class STACK and element is an integer to be pushed on to top of thestack.s1=s1- ; where s1 is an object of the class STACK and – operator pops the element. Handle the STACK Empty and STACK Full conditions. Also display the contents of the stack after each operation, by overloading the operator<<.

6) Write a C++ program to create a class **NAME** and implement the following operations. Display the result after every operation by overloading the<<.

i) NAME firstname =“Herbert”

ii) NAME lastname =“Schield”

iii) NAME fullname = firstname+lastname

(Use copyconstructor)

7) Write a C++ program to create a class called **MATRIX** using a two-dimensional array of integers. Implement the following operations by overloading the operator = = which checks the compatibility of two matrices m1and m2 to be added and subtracted. Perform the addition and subtraction

by overloading the operators + and – respectively. Display the results (sum matrix m3 and difference matrix m4) by overloading the operator <<.

```
if (m1 = = m2)

{

m3 = m1 + m2;

m4 = m1 – m2;

}

Else

Display error.
```

8) Write a C++ program to create a class called **COMPLEX** and implement the following overloading functions ADD that return a COMPLEX number.

i. ADD (a, s2) – where s1 is an integer (real part) and s2 is a complex number.

ii. ADD (s1, s2) – where s1 and s2 are complex numbers.

9) Write a C++ program to exchange two numbers using function overloading.

10) Design three classes called **STUDENT, EXAM and RESULT**. The student class has data members Such as those represent Rollno, Name and Branch etc. Create the class EXAM by inheriting the STUDENT class. The EXAM class adds data members representing the marks scored in six subjects. Derive the RESULT class from the EXAM class and it has its own data members. Such as total_marks. Write an interactive program to model this inheritance relationship.

11) Create classes **RESERVATION, ADULT, SENIOR_CITIZEN,CHILD** .The Reservation class containing data members, Name of passenger, age, date of journey, Source, Destination, Ticket charge. Write an interactive program to display the ticket charges depending upon the category of passenger.

The classes ADULT, SENIOR_CITIZEN, CHILD are the derived class of RESERVATION.

(Note: Category CHILDREN = ½ of adult ticket charge.

Senior_citizen = ¼ of adult ticket charge.)

13) Write a C++ program to demonstrate how a pure virtual function is defined, declared and invoked from the object of a derived class through the pointer of base class.

14) Write a C++ program to perform **QUICKSORT** for N numbers using template function. Demonstrate sorting of integers and doubles.

Question paper pattern:

In SEE, students will be asked to execute the programs which may be related to the above list of programs

Reference: Lab Manual

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	To demonstrate the understanding and application of object oriented programming principles.
	CO2	To show expertise and proficiency in logical decision making / thinking
	CO3	To exhibit the skills of writing programs related to object oriented programming concepts in order to generate the necessary output.
	CO4	To reveal the skill of oral communication to present his /her views on programming aspects.
	CO5	To prepare report about the technical details of experimental work related to application or development.

FOURTH SEMESTER

Mathematical Methods for Information Science

SubjectCode:19MA41 **Credits:04**

CIE:50Marks **SEE:50Marks** **SEE: 3hours**

Hours/Week:3 hours (Theory) **Total: 28hours**

Prerequisite: Basic knowledge Integration, Statistics and probability

Course objectives:

To enable the students to obtain the knowledge of Engineering Mathematics in the following topics

1. Probability distribution of discrete and continuous random variables
2. Joint probability distributions and discrete and continuous random variables and Morkov'schains
3. Analyse the sample data using Large sample test, t-distribution and chi-distribution

Module-I

Probability distributions: **6hours**

Random variable (Discrete and continuous) p.d.f., c.d.f., Binomial distribution, Poisson distributions, Normal distribution and problems

Module-II

Joint probability distributions: **6 hours**

Concept of joint probability distribution, discrete and continuous random variables independent random variables .problems on expectation and variance

Module -III

Markov chains: **5hours**

Introduction probability vectors stochastic matrices, higher transition probability. Stationary distribution of regular Markov chains and absorbing states

Module IV

Sampling theory-I **5hours**

Sampling, sampling distribution, standard error,null and alternative hypothesis,Type-I and Type-II errors, Confidence limits. Test of significance for Large sample: Test for single proportion,difference of proportions, single mean, difference of means, and difference of standard deviations.

Module V

Sampling theory-II **6hours**

Test of significance Small samples student's t-distribution: Test for single mean, difference of means, test for ratio of variances - Chi-square test for goodness of fit and independence of attributes. And problems

CO1: Solve problems using theoretical probability distributions

CO2: Apply the concepts of joint probability, to find covariance, correlation, independent variables

CO3: Apply stochastic to find the probability vectors, stochastic matrices and higher transition probability

CO4: Analyze the sample data using Large sample tests

CO5: Analyze the sample data using t-distribution and chi- distribution

Text book:

1 Higher Engineering Mathematics by B.S.Grewal, 36thEdn.

2 Engineering Mathematics by N. P. Bali and Manish Goyal. Laxmi publications, latest edition.

3 Higher Engineering Mathematics by H. K. Dass and Er. Rajnish Verma. S. Chand publishing 1st edition-2011

4 Statistical Methods Authored By Gupta S.P.Publisher: Sultan Chand &Sons.Publishing Year:2021

5 Fundamentals of Mathematical Statistics Authored By Gupta S.C.& Kapoor V.K.Publisher:Sultan Chand &Sons.Publishing Year:2020

Reference books:

1. Advanced Engineering Mathematics by E. Kreyszig, John Willey & sons 8thEdn.

2. Advanced Engineering Mathematics by R.K.Jain& S.R.K Iyengar; Narosa publishing House.

3. Introductory methods of numerical analysis.

PYTHON PROGRAMMING WITH UNIX SYSTEMS

Subject Code: 19IS42	CREDITS: 4	CIE: 50
Number of Lecture Hours/Week	4 (Theory)	SEE: 50
Total Number of Lecture Hours	52	SEE Hours: 03

Prerequisite: The students should have the basic knowledge of C and C++.

Course Objectives:
To enable the students to obtain the knowledge of Programming Python With Unix Systems to:

- Understand the UNIX environment and its File System.
- Understand Shell scripts, process and administrative privileges.
- Understand the basic principles of Python programs and IDEAL environment.
- Understand the control and loop structures in Python and string and file handling mechanisms.
- Build GUI applications.

Modules	Teaching Hours
Module I **Introduction**- The UNIX Environment, UNIX Structure, Commands, File Systems- Operations on Directories and Regular Files, Security and File Permission **- Vi Editor - The Basic vi Editor and its operations** **Introduction to Shells-** Unix Session , Standard Streams , Redirection, Pipes , tee command, Command execution , Quotes , Command substitution, Job Control, Aliases, Variables, predefined variables, Options, Shell/Environment Customization. **Shell Programming** – Basic Script Concepts, Expressions, Decisions: Making Selections, Repetition, Special Parameters and variables, Changing Positional Parameters, Argument Validation, Debugging Scripts, Script Examples.	**10 Hours**
Module II **The Process in UNIX:** Process Basics, ps: Process Status, System Processes. **UNIX System Administration:** root: The System Administrator's Login, The Administrator's Privileges, Maintaining Security, /etc/passwd and /etc/shadow files, The crypt command, User Management, Start up and Shutdown, Managing Disk Space, Device Files, Handling Floppy Diskettes **Python Programming Language-** About Python, Python development environment, Programming fundamental concepts, Identifiers, Reserved Words, Lines and Indentions, Multi-Line Statements, Command Line Arguments, Variable Types, numbers and strings.	**10 Hours**
Module III Operators, Arithmetic Operator, Comparison Operator, Assignment Operator,	**11Hours**

Bitwise Operator, Logical Operator and Assignment Operator, Decision Making Statements, loop statements. Lists, Built-in-List, Functions and Methods, Functions in lists. Tuples, Built-in-tuples.	
Module IV Methods, Objects and their use, Object oriented programming, Class declaration, Encapsulation, Inheritance, Polymorphism.	**10 Hours**
Module V **Models-**Python models ,name spaces, text files, string processing, Exceptional handling in Python **GUI Programming-**Introduction, Tkinter programming, Designer Attributes, Tkinter widgets, Project Development using Python Modules.	**11 Hours**

Question paper pattern:

1.The question paper will have TEN questions.
2.There will be TWO questions in each module, covering all the topics.
3.The student need to answer FIVE full questions, selecting ONE full question from each module

Text books:

1. Charles Dierbach, Introduction to Computer Science using PYTHON - A Computational Problem -Solving Focus, Wiley India Edition
2. Sumitabha Das, UNIX Concepts and Applications Fourth Edition, Tata McGraw Hill Publications, 2009.

Reference Books:

1. Reference Books: 1. Kenneth A. Lambert , B.L Juneja , “Fundamentals of Python Programming”, Cengage Learning,ISBN:978- 81-315-2903-4, 2015

E books and online course materials:

Course outcomes:
On completion of the course, the student will have the ability to:

Course Code	CO #	Course Outcome (CO)	Blooms Level
	CO1	Demonstrate the working of Unix Operating System and Categorize the concepts of Shell and implement different commands and scripts in shell	**L3**
	CO2	Analyze the process handling mechanism and system administration tasks and Demonstrate the working of Python Programming Principles	**L4**

	CO3	Analyze the working principles of lists, tuples and functions	**L2,L4**
	CO4	Implement Object Oriented Programming Principles in Python	**L3**
	CO5	Implement PY QT Designer to build GUI applications	**L3**

FINITE AUTOMATA AND FORMAL LANGUAGES

Subject Code	**19IS43**	Credits:03
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 3 Hours		Total Hours: 42

Prerequisite: The students should have good knowledge of discrete mathematical structures, data structures, programming principles and computer architecture.

Course Objectives: To enable the students to obtain the knowledge of Finite Automata and Formal languages in the following topics:

- Design automata, regular expressions and context-free grammars accepting or generating a certain language.
- Transform between equivalence deterministic and non-deterministic finite automata and regular expressions.
- Design and translate between deterministic and non-deterministic push down automata.
- Define Turing machines performing simple tasks.

Modules	**Teaching Hours**
Module I **Automata:** Why study automata theory, introduction to formal proof, inductive proofs, the central concept of automata theory. Deterministic Finite Automata, Nondeterministic Finite Automata, Finite Automata with Epsilon Transitions.	**8 Hours**
Module II **Finite automata, regular expressions**: An application of finite automata; Regular expressions; Finite Automata and Regular Expressions; Applications of Regular Expressions. Algebraic Laws for Regular Expressions. **Properties of Regular Languages**: Proving Languages not to be Regular, Closure Properties of Regular Languages, Equivalence and Minimization of Automata.	**8 Hours**
Module III **Context-Free Grammars and Languages**: Context-free Grammars, Parse Trees, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages. **Properties of Context-Free Languages**: Normal Forms for Context-Free Grammars, The Pumping Lemma for Context-free Languages	**9 Hours**

Module IV **Properties of Context-Free Languages**: Closure Properties of Context-Free Languages. **Pushdown Automata**: Definition of the Pushdown Automaton, The Languages of a PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automata.	**9 Hours**
Module V **Introduction to Turing machine:** Problems that Computers cannot solve; The turning machine; Programming techniques for Turning Machines; Extensions to the basic Turing Machines; Turing machines and computers	**8 Hours**

Question paper pattern:

1. The question paper will have TEN questions.
2. There will be TWO questions in each module, covering all the topics.
3. The student need to answer FIVE full questions, selecting ONE full question from each module.

Text books:

Introduction to Automata Theory, Languages and Computation – John E.. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman:, 3rd Edition, Pearson education, 2007.

Reference Books:

1. Fundamentals of the Theory of Computation: Principles and Practice – Raymond Greenlaw, H.James Hoove, Morgan Kaufmann.
2. Introduction to Languages and Automata Theory – John C Martin, 3rd Edition, Tata McGraw- Hill, 2007.
3. Introduction to Computer Theory – Daniel I.A. Cohen, 2nd Edition, John Wiley & Sons, 2004. 4. An Introduction to the Theory of Computer Science, Languages and Machines – Thomas A. Sudkamp, 3rdEdition, Pearson Education,2006.

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Analyze different concepts of automata theory to design DFA and NFA
	CO2	Translate finite automata to regular expressions and vice versa and construct regular expressions for given languages
	CO3	Design context free grammar accepting or generating certain language and prove some sets are not regular.
	CO4	Design and Translate deterministic and non deterministic pushdown automata for a given context free language
	CO5	Identify the limitations of computational models and possible methods of proving them and construct Turing machine.

ANALYSIS AND DESIGN OF ALGORITHMS

Subject Code	**19IS44**	Credits:03
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 03 Hours (Theory)		Total Hours: 42

Prerequisite: The students should have the knowledge of discrete mathematical structures and C programming principles.

Course Objectives: To enable the students to obtain the knowledge of Analysis and Design of Algorithms in the following topics.

- Understand algorithm Design and analysis process.
- Describe various sorting and searching techniques.
- Understand different algorithm design techniques.
- Apply appropriate method to solve a given problem.

Modules	**Teaching Hours**
Module I **Introduction:** What is an Algorithm? Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental Data Structures **Fundamentals of the Analysis of Algorithm Efficiency:** Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non-recursive and Recursive Algorithms, Example -Fibonacci Numbers	**9 Hours**
Module II **Brute Force:** Selection Sort and Bubble Sort, Sequential Search and Brute-Force String Matching, Exhaustive Search **Divide and Conquer:** Merge sort, Quick sort, Binary Search	**8 Hours**
Module III **Divide and Conquer contd**: Binary tree traversals and related properties, Multiplication of large integers and Stressen's Matrix Multiplication .Decrease and Conquer: Insertion Sort, Depth First Search, Breadth First Search, Topological Sorting. **Transform and Conquer:** Balanced Search Trees, Heaps and Heap sort, Space and Time Tradeoff : Input Enhancement in String Matching.	**8 Hours**

Module IV **Space and Time Tradeoff** Contd: Hashing Dynamic Programming: Computing a Binomial Coefficient, Warshall's and Floyd's Algorithms, the Knapsack Problem and Memory Function	**8 Hours**
Module V **Greedy Technique:** Prim's Algorithm, Kruskal’s Algorithm, Dijkstra's Algorithm, Huffman Trees **Limitations of Algorithm Power:** Lower-Bound Arguments, Decision Trees **Limitations of Algorithm Power** contd: P, NP and NP-Complete Problems. Coping with the Limitations of Algorithm Power: Backtracking, Branch-and Bound, Approximation Algorithms NP-Hard Problems	**9 Hours**

Question paper pattern:

1. The question paper will have TEN questions.
2. There will be TWO questions in each module, covering all the topics.
3. The student need to answer FIVE full questions, selecting ONE full question from each module.

Text books:

Text Book: Introduction to The Design & Analysis of Algorithms, Anany Levitin. 3rd Edition, Pearson Education, 2008.

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal k. Rivesk Clifford Stein, 2ndEdition, PHI,2006.
2. Computer Algorithms by Horowi'tzE., Sahni S., RajasekaranS.,Galgofia Publications

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Analyze the performance of algorithms.
	CO2	Identify the given problem and design the algorithm.
	CO3	Implement Searching, Sorting and Graph Traversal Algorithms.
	CO4	Analyze deterministic and Non-deterministic completeness and identify different NP problems.
	CO5	Design and analyze algorithms using Greedy, Backtracking, Branch & Bound techniques.

MICROPROCESSORS AND ARM PROCESSORS		
Subject Code	**19IS45**	Credits:04
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 04 Hours (Theory)		Total Hours: 52

Prerequisite: The students should have an understanding of Logic Design and Computer Organization.

Course Objectives: To enable the students to obtain the knowledge of Microprocessors and ARM processors in the following topics.

- Become familiar with the architecture and the instruction set of an Intel microprocessor
- Design and develop Assembly language programs
- Study the design of various types of peripheral interfacing
- Understand the architecture and programming of ARM Microprocessor.

Modules	**Teaching Hours**
Module I **The Microprocessor and its Architecture:** Introduction, Internal microprocessor architecture, Real mode memory addressing, protected mode memory addressing, memory paging. **Addressing Modes:** Data Addressing Modes, Program memory-Addressing Modes, Stack memory addressing modes. **8086/8088 Hardware Specifications:** Pin-outs and the pin function, clock generator, bus buffering and latching, bus timing, ready and the wait state, minimum mode versus maximum mode	**11 Hours**
Module II **Data Movement Instructions:** MOV revisited, PUSH/POP, Load effective address, string data transfers, miscellaneous data transfer instructions.	

Arithmetic and Logic Instructions: Addition, Subtraction and comparison, Multiplication and division, BCD and ASCII Arithmetic, Basic logic instructions, shift and rotate, string comparisons.	**10 Hours**
Module III **Program Control Instructions:** The jump group, controlling the flow of an assembly language program, procedures, Introduction to interrupts, machine control and miscellaneous instructions. **Programming the Microprocessor:** Modular programming using the keyboard and video display, data conversions, disk files, example programs.	**10Hours**
Module IV **Memory Interface:** Memory devices, Address decoding, 8088 and 80188(8 bit) Memory interface, 8086 (16-Bit) memory interface. **Basic I/O Interface:** Introduction to I/O interface, I/O port address decoding, The programmable peripheral interface 82C55, programmable keyboard/Display interface 8279, programmable interval timer 8254	**10 Hours**
Module V **Introduction to ARM**: RISC and CISC Architectures T**he ARM Architecture**: The Acorn RISC Machine, **The ARM programmer's model**: General purpose registers, CPSR, SPSR, ARM memory map, data format, load and store architecture, ARM development tools. ARM Assembly language Programming	**11 Hours**
Question paper pattern: 1.The question paper will have TEN questions. 2.There will be TWO questions in each module, covering all the topics. 3.The student need to answer FIVE full questions, selecting ONE full question from each module.	
Text books: 1. The Intel Microprocessors 8086/8088, 80186/80188, 80286, 80486, Pentium and Pentium pro processor by Barry B. Brey 8th Edition, Pearson Education. 2013 **2.** ARM System-on-Chip Architecture, Steve Furber, Second Edition, Pearson, 2015	
Reference Books: 1. Advanced Microprocessors and peripherals Architecture, Programming and Interfacing by Ajay Kumar Ray & Kishore MBurchandi 2. Microprocessor architecture, Programming and Applications with the 8085 by Ramesh S Goankar,4th edition 3. MicroprocessorsandInterfacingProgrammingandHardware,DouglasV.Hall,2ndEdition,Tata	

McGraw Hill.
4. The Definitive Guide to the ARM Cortex-M3, by Joseph Yiu, 2nd Edition , Newnes, 2009

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Demonstrate the architecture and organization of the processor.
	CO2	Develop assembly language programs using arithmetic, logical and data movement instructions
	CO3	Design and develop the application programs with use of subroutines & interfacing the input output devices.
	CO4	Formulate interrupt programs for memory systems using different interrupt methods and interface I/O devices.
	CO5	Design the internal architecture of ARM processor and develop program for ARM processor

ENVIRONMENTAL STUDIES

Subject Code:	**19CV46**	Credits:00
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 02 Hours (Theory)		Total Hours: 14

Prerequisite: The students should have the knowledge of discrete mathematical structures and C programming principles.

Course Objectives:

To enable the student for acquiring the knowledge in the following topics.

- Understanding of ecosystem and describing the effects of human activities on environment
- Explaining the importance of water, consequences of contamination of water and the importance of bio geochemical cycles
- Different types of energy sources and environmental impact assessment
- Types of environmental pollution and their effects
- Legal aspects related to environmental pollution and role of Government and NGOs in environmental protection

Modules	**Teaching Hours**

Module I **Environment**-Definition, Ecosystem-Balanced Ecosystem, Human **Activities-**Food Shelter, economic and Social Security.	**06 Hours**
Module II **Effects of human activities on Environment**-Agriculture, Housing, Industry, Mining and Transportation activities. **Natural Resources-** Water **Resources-** Agriculture and quality aspects. Water borne diseases, Water induced diseases, fluoride problems in drinking water, mineral resources, forest wealth, and material cycles-Carbon, Nitrogen and Sulphur cycles.	**06 Hours**
Module III **Energy-** Different types of energy, Electromagnetic radiations, conventional and non conventional sources-Hydro Electric, Fossil fuel based, Nuclear. Biomass and Bio-gas. Hydrogen as an alternative future source of energy. Environmental impact assessment. Sustainable development	**06 Hours**
Module IV Environmental pollution and their effects. Water pollution, Land pollution, Noise pollution, public health aspects. Current Environmental issues and importance: Population Growth, Climate change and global warning effects, Urbanization, Automobile pollution.	**05 Hours**
Module V Acid Rain, Ozone layer depletion, Animal Husbandry. Environmental protection-Role of Government, legal aspects, Initiatives by Non-Government Organizations (NGO) Environmental Education, Womens Education.	**05 Hours**

Question paper pattern:

1. The question paper will have TEN questions.
2. There will be TWO questions in each module, covering all the topics.
3. The student need to answer FIVE full questions, selecting ONE full question from each module.

Text books:

1. Environmental Studies- Benny joseph-TataMcgrawhill-2005.
2. Environmental Studies-Dr. D L Manjunath, PearsonEducation-2006

Reference Books:

1. Principals of Environmental science and Engineering-P.Venugopala Rao Prentice hall ofIndia
2. Environmental Science and Engineering-Meenakshi, Prentice HallIndia

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	CO #	Course Outcome (CO)
	CO1	Describe ecosystem, environment, and food, shelter, explain social and economic security.
	CO2	Explain the effects of human activities on environment, describe the natural resources and bio geochemical Cycle
	CO3	Describe the impacts of energy sources, importance of non-conventional energy sources and sustainable Development
	CO4	Explain the causes, effects, and remedial measures of land pollution, water pollution, noise pollution and current environmental problems
	CO5	Explain the causes and effects of global environmental pollution, role of government and non government organization in environmental protection and importance of environmental education and women’s education.

BALAKE KANNADA

Subject Code	Subject	Stream	Th– Tut-Pr	Credits
19KAK37/19KAN47	BALAKE KANNADA	Humanities and Social Sciences (H.S.S)	2 - 0 - 0	01

CIE:50 SEE: SEE : 1 hours30Minutes Total : 28Hours

ವಿಶ್ವೇಶ್ವರಯ್ಯ ತಾಂತ್ರಿಕ ವಿಶ್ವವಿದ್ಯಾಲಯ
ಬಳಕೆ ಕನ್ನಡ ಪಠ್ಯಪುಸ್ತಕ

baLake Kannada Text Book for VTU

(Common to B.Arch, B.Plan and B.E/B.Tech of all branches)

[As per Outcome Based Education (OBE) and Choice Based Credit System (CBCS) scheme]

Course Learning Objectives:

The course will enable the non Kannadiga students to understand, speak, read and write Kannada language and communicate (converse) in Kannada language in their daily life with kannada speakers.

Table of Contents

Part – I Lessons to teach and Learn Kannada Language

arid Plural markers

Lesson – 7 ನ್ಯೂನ / ನಿಷೇಧಾರ್ಥಕ ಕ್ರಿಯಾಪದಗಳು ಮತ್ತು ಗುಣವಾಚಕಗಳು
Defective.." Negative 3 erbs arid C oloiir .Adjectives

Lesson – 8 ಅಪ್ಪಣೆ / ಒಪ್ಪಿಗೆ, ನಿರ್ದೇಶನ, ಪ್ರೋತ್ಸಾಹ ಮತು ಒತ್ತಾಯ ಆರ್ಥರೂಪ
mind ss›d zao6,rt6-Permission,Commands,encouraging
and Urging words (Imperative words andsentences)

Lesson – 9 ಸಾಮಾನ್ಯ ಸಂಭಾಷಣೆಗಳಲ್ಲಿ ದ್ವಿತೀಯ ವಿಭಕ್ತಿ ಪ್ರತ್ಯಯಗಳು ಮತ್ತು
ño e4fcd› &aadñu
Accusative Cases and Potential Forms used in General Communication

Lesson – 10 “ಇರು ಮತ್ತು ಇರಲ್ಲ” ಸಹಾಯಕ ಕ್ರಿಯಾಪದಗಳು, ಸಂಭಾವ್ಯಸೂಚಕ ಮತ್ತು

Helping Verbs 'iru and iralla”. Corresponding Future and
Negation Verbs

Lesson – 11 ಹೋಲಿಕೆ (ತರತಮ) , ಸಂಬಂಧ ಸೂಚಕ ಮತ್ತು ವಸ್ತು ಸೂಚಕ
ಪ್ರತ್ಯಯಗಳು ಮತ್ತು ನಿಷೇಧಾರ್ಥಕ ಪದಗಳ ಬಳಕೆ
Comparitive, Relationship. Identification and Negation Words

Lesson—12 aac› zzlcda fizzlcccb6 osoña 6 cd›ozdclrt'" ‹›3‹›3@ z aa dñ•l›
Different types of forms of Tense, Time and Verbs

Lesson—13 0, -@, - A3, - @¥3, - esR, - csm , - W, -a°, 'a6,6/cdza
z ¥tccbñ«2ao6rt azl, ‹›3zslt&zzlcda s2zlvz zazdaac› zaoz/6clzdzd
Formation of Past, Future and Present Tense Sentences with
Verb Forms

Lesson — 17 PART - II
Kaniiaila Language ficritit Part — 1

Lesson – 18 PART - III
Kannada Vocabulary List : ಸಂಭಾಷಣೆಯಲ್ಲಿ ದಿನೋಪಯೋಗಿ ಕನ್ನಡ
Afi d'th - Kannada 1\"ords in C.onversation

B.E. IV SEM.
SAMSKRUTHIKA
KANNADA

Subject Code	Subject	Stream	Th– Tut-Pr	Credits
19KAK47	SAMSKRUTHIKA KANNADA	Humanities and Social Sciences (H.S.S)	2 - 0 - 0	01

CIE:50 SEE: SEE : 1 hours30Minutes Total : 28Hours

ಸಾಂಸ್ಕೃತಿಕ ಕನ್ನಡ ಪಠ್ಯಪುಸ್ತಕ

(ಕನ್ನಡ ಮಾತೃಭಾಷೆಯ ವಿದ್ಯಾರ್ಥಿಗಳಿಗೆ)

(ಕನ್ನಡಿಗರಿಗಾಗಿ – **for Kannadigas** - Common to all branches)

[As per Outcome Based Education (OBE) and Choice Based Credit System (CBCS) scheme]

ಸಾಂಸ್ಕೃತಿಕ ಕನ್ನಡ ಕಲಿಕೆಯ ಉದ್ದೇಶಗಳು:

- ಪದವಿ ವಿದ್ಯಾರ್ಥಿಗಳಾಗಿರುವುದರಿಂದ ಸಾಂಸ್ಕೃತಿಕ ಕನ್ನಡದ ಜೊತೆಗೆ ಕ್ರಿಯಾತ್ಮಕ ಕನ್ನಡವನ್ನು, ಕನ್ನಡ ಸಾಹಿತ್ಯ, ಸಂಸ್ಕೃತಿ ಮತ್ತು ನಾಡು ನುಡಿಯ ಪರಿಚಯ ಮಾಡಿಕೊಡುವುದು.
- ಕನ್ನಡದಲ್ಲಿ ತಾಂತ್ರಿಕ ವಿಜ್ಞಾನಗಳ ವಿಷಯಕ್ಕೆ ಸಂಬಂಧಿಸಿದ ಹಲವಾರು ವಿಷಯಗಳನ್ನು ಪರಿಚಯ ಮಾಡಿಕೊಡುವುದು.
- ಕನ್ನಡ ಭಾಷಾಭ್ಯಾಸ, ಸಾಮಾನ್ಯ ಕನ್ನಡ ಹಾಗೂ ಆಡಳಿತ ಕನ್ನಡದ ಪದಗಳ ಪರಿಚಯ ಮಾಡಿಕೊಡುವುದು.

ಪರಿವಿಡಿ

ಭಾಗ – ಒಂದು ಲೇಖನಗಳು

ಕನ್ನಡ ನಾಡು, ನುಡಿ ಮತ್ತು ಸಂಸ್ಕೃತಿಗೆ ಸಂಬಂಧಿಸಿದ ಲೇಖನಗಳು

೧. ಕರ್ನಾಟಕ ಸಂಸ್ಕೃತಿ : ಹಂಪ ನಾಗರಾಜಯ್ಯ

೨. ಕರ್ನಾಟಕದ ಏಕೀಕರಣ : ಒಂದು ಅಪೂರ್ವ ಚರಿತ್ರೆ – ಜಿ. ವೆಂಕಟಸುಬ್ಬಯ್ಯ

೩. ಆಡಳಿತ ಭಾಷೆಯಾಗಿ ಕನ್ನಡ – ಡಾ. ಎಲ್. ತಿಮ್ಮೇಶ ಮತ್ತು ಪ್ರೊ. ವಿ. ಕೇಶವಮೂರ್ತಿ *

ಭಾಗ – ಎರಡು

ಕಾವ್ಯ ಭಾಗ (ಆಧುನಿಕ ಪೂರ್ವ)

೪. ವಚನಗಳು : ಬಸವಣ್ಣ, ಅಕ್ಕಮಹಾದೇವಿ, ಅಲ್ಲಮಪ್ರಭು, ಆಯ್ದಕ್ಕಿ ಮಾರಯ್ಯ,
ಜೇಡರ ದಾಸಿಮಯ್ಯ, ಆಯ್ದಕ್ಕಿ ಲಕ್ಕಮ್ಮ.

೫. ಕೀರ್ತನೆಗಳು : ಅದರಿಂದೇನು ಫಲ ಇದರಿಂದೇನು ಫಲ – ಪುರಂದರದಾಸ
ತಲ್ಲಣಿಸದಿರು ಕಂಡ್ಯ ತಾಳು ಮನವೆ – ಕನಕದಾಸ

೬. ತತ್ವಪದಗಳು : ಸಾವಿರ ಕೊಡಗಳ ಸುಟ್ಟು – ಶಿಶುನಾಳ ಷರೀಫ
ಶಿವಯೋಗಿ – ಬಾಲಲೀಲಾ ಮಹಾಂತ ಶಿವಯೋಗಿ

<table>
<tr><td colspan="3"><b>PYHTON AND UNIX SHELL PROGRAMMING LAB</b></td></tr>
<tr><td>Subject Code</td><td><b>19ISL41</b></td><td>Credits:01</td></tr>
<tr><td>CIE : 50</td><td>SEE : 50</td><td>SEE : 03 Hrs</td></tr>
<tr><td colspan="2">Hours / Week : 02Hours(practical)</td><td>Total Hours:14</td></tr>
<tr><td colspan="3"><b>Prerequisite:</b> The students should have the basic knowledge of C programming, command line usage and object oriented programming principles.</td></tr>
<tr><td colspan="3"><b>Course Objectives:</b> To enable the students to Obtain
• The knowledge of Python With Unix Shell Programming.
• The Basic UNIX general purpose Commands and shell scripts.
• The knowledge of file management and permission commands, awk and perl scripts.
• The knowledge to Create and execute Python programs and read data from a text file using Python. GUI Programming in Python.</td></tr>
<tr><td colspan="2"><b>Experiments</b></td><td></td></tr>
<tr><td colspan="2"><b>UNIX System Programming</b>
1. Shell script that accepts two filenames as arguments. Checks if the permissions for these files are identical and if the permissions are identical, outputs the common permissions, otherwise outputs each filename followed by its permission.
2. Shell function that takes a valid directory names as an argument and recursively descends all the subdirectories finds the maximum length of any file in that hierarchy and writes this maximum value to the standard output.
3. Shell Script to display the calendar for current month with current date replace by * or ** depending on whether date has one digit or two digits.
4. Shell script to implement terminal locking. It should prompt the user for a</td><td></td></tr>
</table>

password. After accepting the password entered by the user it must prompt again for password confirmation (to retype the password). If a match occurs, it must lock the terminal and prompt for the password. If the proper password is entered, the terminal must be unlocked. Note the script must be written to disregard BREAK control-D etc. No time limit need be implemented for the lock direction.

5. write a shell script that will take an input file and remove identical lines (or duplicate lines from the file)

6. Write a shell script sum.sh that takes an unspecified number of command line arguments (up to 9) of int s and finds their sum. Modify the code to add a number to the sum only if the number is greater than10

PYTHON

1.Write a program that prompts the user to enter a 5-letter word, and then

prints Extract the first, third, and fifth letters of that word.

Extract the second letter.

Extract the first four letters.

Extract the last six letters.

2.Write a python program that allows us to determine the length of the time needed to pay off a credit card balance, as well as the total interest paid.

3.Write a program that takes in a number as input and check whether the given number is prime or not and also generate the first 50 prime numbers.

4.Write a program that displays the details of the employees and total number of employees created using class and objects.

5.Write a python program that counts the number of objects created and also prints their respective id’s.

6.Write a python program that implements the concept of multilevel inheritance using qt or pyqt designer.

7.Write a python program that illustrates the multiple inheritance using qt or pyqt designer.

8.Write python program using qt or pyqt designer that creates a root window with different widgets and also apply actions to the widgets using inbuilt functions.

9.Write python program using qt or pyqt designer that creates a root window and the label is dynamically incremented by 1 until a stop button is pressed

10.Design students GUI with their attributes like stud_id, name, dept, courses_offered, course_credits. Connect this with the data base that is created using mysql.

SEE Question Paper Pattern:

Students will be asked to execute two programs from Unix System Programming and One from Python

Reference: Lab Manual

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Apply basic UNIX general purpose commands and shell scripts
	CO2	Analyze the file management and permission commands.
	CO3	Design and Implement simple modules using shell, pearl and awk scripts.
	CO4	Implement simple programs using python programming principles.
	CO5	Design GUI using PyQt Designer tool.

ALGORITHMS LAB		
Subject Code	**19ISL42**	Credits:01
CIE : 50	SEE : 50	SEE : 03 Hrs
Hours / Week : 02 Hours(Practical)		Total Hours:14
Prerequisite: The students must have the knowledge of C, C++ programming concepts and usage of summation formulae, recurrences in mathematics.		
Course Objectives: To enable the students to • obtain the knowledge of Algorithms • Understand different search and sort techniques • Understand the binary tree principles • Understand the different algorithms to solve the problems.		
Experiments		**Teaching Hours**
IMPLEMENT THE FOLLOWING USING C LANGUAGE: 1. Implement Recursive Binary search and Linear search and determine the time required to search an element. 2. Sort a given set of elements using Heap sort method and determine the time required to sort the elements. 3. Sort a given set of elements using Merge sort method and determine the time required to sort the elements. 4. Sort a given set of elements using Selection sort and determine the time required to sort elements. 5. Implement 0/1 Knapsack problem using dynamic programming. 6. From a given vertex in a weighted connected graph, find shortest Paths to other vertices using Dijkstra's algorithm. 7. Sort a given set of elements using Quick sort method an determine the time required to sort the elements. 8. Find Minimum Cost Spanning Tree of a given un directed graph using Kruskal's algorithm. 9. 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. 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{l,2,6}and{l,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.

11. A. Implement Hors pool algorithm for String Matching.
    B. Find the Binomial Co-efficient using Dynamic Programming.
12. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
13. A. Implement Floyd’s algorithm for the All-Pairs-Shortest-Paths Problem.
    B. Compute the transitive closure of a given directed graph using Warshall's algorithm.
14. Implement N Queen's problem using Back Tracking.

Question paper pattern:

In SEE, students will be asked to execute one program which may be related to the above list of programs.

Reference: Lab Manual

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Identify the problem given and design the algorithm using various design techniques
	CO2	Design and implement basic data structure for searching and sorting algorithm
	CO3	Describe the advanced sorting and graph algorithm
	CO4	Illustrate concepts of computational complexity and computability and be able to apply in practice.
	CO5	Compare the performance of different algorithms for same problem.

MICROPROCESSORS AND ARM LAB		
Subject Code	**19ISL43**	Credits:01
CIE: 50	SEE: 50	SEE: 03 Hrs
Hours / Week: 02 Hours		Total Hours:14

Prerequisite: NIL

Course Objectives:

To enable the students to obtain the knowledge of Microprocessors and Arm Lab in the following topics.

- To become familiar with the architecture and Instruction set of Intel 8085microprocessor.
- To provide practical hands on experience with Assembly Language Programming.
- To familiarize with interfacing of various peripheral devices with 8085microprocessor.

Experiments	**Teaching Hours**

SOFTWARE PROGRAMS: PART A

1. Design and develop an assembly language program to search a key element “X” in a list of ‘n’16-bit numbers. Adopt Binary search algorithm in your program for searching.
2. Design and develop an assembly program to sort a given set of ‘n’ 16-bit numbers in ascending order. Adopt Bubble Sort algorithm to sort given elements.
3. Develop an assembly language program to reverse a given string and verify whether it is a palindrome or not. Display the appropriate message.
4. Develop an assembly language program to compute nCr using recursive procedure. Assume that ‘n’ and ‘r’ are non-negative integers.
5. Design and develop an assembly language program to read the current time and Date from the system and display it in the standard format on the screen.
6. To write and simulate ARM assembly language programs for data transfer, arithmetic and logical operations (Demonstrate with the help of a suitable program).
7. To write and simulate C Programs for ARM microprocessor using KEIL (Demonstrate with the help of a suitable program)

HARDWARE PROGRAMS: PART B

8. Design and develop an assembly program to demonstrate BCD Up-Down Counter (00-99) on the Logic Controller Interface.

9. Design and develop an assembly program to read the status of two 8-bit inputs (X & Y) from the Logic Controller Interface and display X*Y.

10. Design and develop an assembly program to 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).

11. Design and develop an assembly program to drive a Stepper Motor interface and 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).

12. Design and develop an assembly language program to Generate the Sine and Wave and Half Rectified Sine Waveform using DAC interface (The output of the DAC is to be displayed on the CRO).

13. To interface LCD with ARM processor-- ARM7TDMI/LPC2148. Write and execute programs in C language for displaying text messages and numbers on LCD.

14. To interface Stepper motor with ARM processor--ARM7TDMI/LPC2148. Write a program to rotate stepper motor

Question paper pattern:

- All laboratory experiments (all 7 + 7 nos) are to be included for practical examination.

- Students are allowed to pick one experiment from each of the lot.

- Strictly follow the instructions as printed on the cover page of answer script for breakup of marks

- PART –A: Procedure + Conduction + Viva: **04 + 15 +06(25)**

- PART –B: Procedure + Conduction + Viva: **04 + 15 +06(25)**

- Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

Reference: Lab Manual

Course outcomes:

On completion of the course, the student will have the ability to:

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Summarize 80x86 instruction sets and comprehend the knowledge of how assembly language works.
	CO2	Design and develop assembly programs using 80x86 assembly language instructions
	CO3	Infer functioning of hardware devices and interfacing them to x86 family
	CO4	Choose processors for various kinds of applications.
	CO5	Formulate and test ARM programs and also to interface with I/O devices