Centenary Celebrated Sharnbasveshwar Vidya Vardhak Sangha's

Sharnbasva University

ಶರಣಬಸವ ವಿಶ್ವವಿದ್ಯಾಲಯ

A Private University enacted by Govt. of Karnataka as "Sharnbasva University Act 2012" Karnataka Act No. 17 of 2013. Notification No. ED 144 URC 2016 dated 29/07/2017

www.sharnbasvauniversity.edu.in Kalaburagi-585 103 - Karnataka - India Email : sharnbasvauniversity@gmail.com

Faculty of Engineering and Technology (Exclusively for Women)
Department of Computer Science and Engineering
B. Tech 2nd year (III and IV Semester)
Scheme of Teaching and Examination

Outcome Based Education (OBE) & Choice Based Credit System (CBCS)

(Effective from the academic year 2018-19)

Vision & Mission of Faculty of Engineering and Technology (Exclusively for Women)

INSTITUTE VISION MISSION

VISION

- We aspire to become a global model for women professional through quality education and ethical values in the field of Engineering and Technology

MISSION

- The mission of our faculty is to inspire a research culture, encourage entrepreneurial efforts and empower globally to be great leaders.
- To create technical women's power to meet the current and future demands of the industry.
- To develop women professionals who are academically and technically sound with strong ethics and above all, good human beings.

VISION & MISSION OF DEPARTMENT

VISION

- Aspire to become a centre of excellence for quality technical education and research by keeping pace with new technologies to empower girl students to lead and excel in the field of computer science & engineering along with ethical principles & a sense of social responsibility.

MISSION

- To impart academic excellence, encourage research and innovation in Computer science and engineering.
- To educate the students with knowledge and skills, encourage students to address societal problems through IT solutions.
- To prepare students to develop entrepreneurship skills with proper ethical values and a desire to pursue life-long learning.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs) of the Department

PEO1 PEO 1: Graduates will possess a strong foundation in Computer Science and Engineering that are required for problem solving to excel and succeed in their profession.

PEO2 PEO 2: Graduates will have scientific and engineering breadth to comprehend, analyze, design and solve real life problems using the acquired skills and life long learning

PEO3	PEO 3: Graduates will have exposure to emerging cutting edge technologies and adequate training with opportunity to work on multidisciplinary projects.
PEO4	PEO 4: Graduates will be professional with ethical attitude,effective communication skills, teamwork capability and relate engineering issues to braoder social context

PROGRAMME OUTCOMES (POs) COMPUTER SCIENCE AND ENGINEERING (UG)

PO1 - Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2 - 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.

PO3 - 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.

PO4 - Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5 - Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6 - The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7 - 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.

PO8 - Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9 - Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10 - Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11 - Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments

PO12 - Life-long learning: Recognize the need for, and have the preparation and ability to engage independent and life-long learning in the broadest context of technological change.

PROGRAMME SPECIFIC OUTCOMES (PSOs)

On completion of the B. Tech. (Computer Science & Engineering) degree the graduates will be able to

PSO1: Apply principles of basic sciences and Engineering fundamentals in the field of Computer Science and Engineering

PSO2: Apply computational, algorithmic, and programming skills to implement solutions for real-life problems in diverse domain adapting to emerging technologies through lifelong learning

PSO3: Develop practical abilities, ethical understanding, effective communication and leadership skills for successful careers in industry or academia

Sharnbasva University, Kalaburagi
Scheme of Teaching and Examination 2018-19
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
(Effective from the academic year 2018-19)

III SEMESTER B. Tech.

Computer Science & Engineering

Course Code	Course Title	Teaching Dept. & Paper Setting	T	Credits

				Duration n hours	CIE Mark s		
BSC	18MAT31	Engineering Mathematics-III		3	50		100
PCC	18CS32	Data structures in C and Applications		3	50		100
PCC	18CS33	Electronics Circuits and Logic Design		3	50		100
PCC	18CS34	Computer Organization and Architecture		3	50		100
PCC	18CSL35	Data Structures Lab	2	3	50		100
PCC	18CSL36	Electronics Circuits and Logic Design Lab	2	3	50		100
PCC	18CSL37	UNIX Shell Programming Lab	2	3	50		100
PRJ	18CSP38	Project – III	2	3	50		100
HSMC	18KANKK310/ 20KANAK310	Kannada Kali-3/Ayda Kathegalu		2	50		100

12	5	08	900	

C-
l

d

J-

Sharnbasva University, Kalaburagi

Scheme of Teaching and Examination 2018-19

Outcome Based Education (OBE) and Choice Based Credit System (CBCS)

(Effective from the academic year 2018-19)

IV SEMESTER B. Tech.

Computer Science & Engineering

Course Code	Course Title	Teaching Dept. & Paper Setting Board	Teaching Hours/week			Examination				Credits
			L	T	P	Duration in hours	CIE Marks	SEE Marks	Total Marks	

	BSC	18MAT41	Engineering Mathematics-IV	Mathematics	3	1		3	50	50	100	
	PCC	18CS42	Design and Analysis of Algorithms	CSE	3	1		3	50	50	100	
	PCC	18CS43	Microprocesso r	CSE	3	1		3	50	50	100	
	PCC	18CS44	Java Programming	CSE	3	1		3	50	50	100	
	PCC	18CSL45	Microprocesso r Lab	CSE			2	3	50	50	100	
	PCC	18CSL46	Java Programming lab	CSE			2	3	50	50	100	
	PCC	18CSL47	Algorithm Analysis and Design Lab	CSE			2	3	50	50	100	
	PRJ	18CSP48	Project-IV	CSE			2	3	50	50	100	
	HSMC	18KANAKK4 10/ 20KANMD41 0	Kannada Kali-4/Mahadasohi	Humanities		1		2	50	50	100	
	12	**5**	**08**	**26**	**450**	**450**	**900**	**21**				

BSC-Basic Science, PCC-Professional Core, HSS-Humanity and Social Science, PRJ-Project

Engineering Mathematics-III

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18MAT31	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 04

Course Objectives: This course will enable students to

- To impart the basic concepts of data structures and algorithms.
- To understand concepts about searching and sorting techniques
- To understand basic concepts about stacks, queues, lists, trees and graphs.
- To enable them to write algorithms for solving problems with the help of fundamental data structures

Module I	Hours
Laplace transforms: definition, transforms of elementary functions, properties, periodic function, unit step function, unit impulse function. **Inverse laplace transforms:** definition, convolution theorem (without proof), finding inverse laplace transform by convolution theorem. Solution of linear differential equations using laplace transforms and applications	10
Module II	
Z- **transforms:** difference equations, basic definitions, damping rule, shifting rule, initial and final value theorems (without proof) and problems. Inverse z-transforms. Applications of z-transforms to solve difference equation .	10
Module III	
Statistical methods: correlation-Karl Pearson's co- efficient of correlation problems. Regression analysis lines of regression (without proof)-problems. **curve fitting:** curve fitting by the method of least square. fitting of the curves of the form [illegible] = [illegible] + [illegible], [illegible] = [illegible] + [illegible] + [illegible] & [illegible] = [illegible] **Numerical methods:** numerical solution of algebraic and transcendental equations by Regula -Falsi Method and Newton-Raphson method	10
Module IV	
Finite difference: forward and backward differences, newton's forward and backward interpolation formulae. Divided difference-newton's divided difference formulae. Lagrange's- interpolation formula and inverse interpolation formula (all formula without proof) problems.	10
Module V	
Probability Distribution: Random variables (discrete and continuous) probability mass/density functions. Binomial distribution, Poisson distribution. Exponential and Normal distributions. Problems.	08

Course Outcomes (COs):

CO1	Apply the knowledge of Laplace transform from time domain to frequency domain in Signal and image processing and to find inverse Laplace transform.
CO2	Apply the knowledge of Z-transforms in solving the difference equation arising in the time signals and digital processing.
CO3	Apply the concept of correlation and regression lines for solving the problems and numerical techniques to solve engineering problems.
CO4	Understanding the concepts of Finite differences to solve the problems on interpolation and numerical integration.
CO5	Learn to solve the random variable in both discrete and continuous and their probability distribution, Mass on various engineering problems.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-
CO2	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-
CO3	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-
CO4	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-
CO5	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1.B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, 43rd Ed., 2015.

2. E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons,10th Ed., 2015.

Reference Books:

1. Reference Books: 1. N.P.Bali and Manish Goyal: A Text Book of Engineering Mathematics, Laxmi Publishers , 7th Ed., 2010.
2. 2. B.V.Ramana: "Higher Engineering Mathematics" Tata McGraw-Hill, 2006.
3. 3. H. K. Dass and Er. Rajnish Verma: "Higher Engineering Mathematics", S. Chand publishing, 1st edition, 2011

DATA STRUCTURES IN C AND APPLICATIONS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18CS32	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 04

Course Objectives: This course will enable students to

- To impart the basic concepts of data structures and algorithms.
- To understand concepts about searching and sorting techniques
- To understand basic concepts about stacks, queues, lists, trees and graphs.
- To enable them to write algorithms for solving problems with the help of fundamental data structures

Module I	Hours
Introduction: Data Structures, Data structure Operations, Pointers and Dynamic Memory Allocation, Data Abstraction. **Arrays and structures:** dynamic allocated arrays ,structures & unions, Polynomials and Sparse Matrices. **Array Operations:** Traversing, inserting, deleting, searching, and sorting. **Strings**: Basic Terminology, Storing, Programming Examples.	10
Module II	
Stacks & Queues: Stacks Definition, Stack Operations, Array Representation of Stacks, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression, **Recursion**: Factorial, Fibonacci Sequence, Tower of Hanoi, Ackerman's function. **Queues:** Definition, Representation-array & linked representation of queues. Queue Operations, Circular Queues, Dequeues, Priority Queues	10
Module III	
Linked Lists: Definition, Representation of linked lists in Memory, Memory allocation; Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists and header linked lists. Linked Stacks and Queues. Applications of Linked lists – Polynomials, Additional list operations-inverting singly linked list, concatenating singly linked list. Sparse matrix representation.	08
Module IV	
Trees: Definition, Representation of trees, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations-copying binary tree, testing equality. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching. .	10
Module V	
Graphs: Definitions, Terminologies, Matrix and Adjacency List Representation Of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search. **Sorting and Searching:** Insertion Sort, Radix sort, selection sort. **Hashing**: Hash Table organizations, Hashing Functions.	10

Course Outcomes (COs):

CO1	Acquire the fundamental knowledge on various data structures operations.
CO2	Apply stack and queue data structures in problem solving.
CO3	Analyze linked list for different applications.
CO4	Develop solutions using trees to model the real-world problem.
CO5	Analyse graph structures and hashing techniques to map the data.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	-	-	-	-	-	-	-	-	-	-	1	1	3	-
CO2	3	2	2	-	-	-	-	-	-	-	-	1	1	3	-
CO3	3	2	2	-	-	-	-	-	-	-	-	1	1	3	-
CO4	3	2	2	-	-	-	-	-	1	1	1	1	1	3	-
CO5	3	2	2	-	-	-	-	-	-	-	-	1	1	3	-

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Fundamentals of Data Structures in C - Ellis Horowitz and Sartaj Sahni, 2nd edition, Universities Press, 2014.
2. Data Structures - Seymour Lipschutz, Schaum's Outlines, Revised 1st edition, McGraw Hill, 2014

Reference Books:

1. Data Structures: A Pseudo-code approach with C –Gilberg & Forouzan, 2nd edition, Cengage Learning, 2014.
2.Data Structures using C, Reema Thareja, 3rd edition Oxford press, 2012.
3.An Introduction to Data Structures with Applications- Jean-Paul Tremblay & Paul G. Sorenson, 2nd Edition, McGraw Hill, 2013.
4.Data Structures using C - A M Tenenbaum, PHI, 1989.Data Structures and Program Design in C - Robert Kruse, 2nd edition, PHI, 1996.

ELECTRONICS CIRCUITS AND LOGIC DESIGN

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18CS33	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50

Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 04

Course Objectives: This course will enable students to

- Recall and Recognize construction and characteristics of JFETs and MOSFETs and differentiate with BJT
- Demonstrate and Analyze Operational Amplifier circuits and their applications
- Describe, Illustrate and Analyze Combinational Logic circuits, Simplification of Algebraic Equations using Karnaugh Maps and Quine McClusky Techniques.
- Describe and Design Decoders, Encoders, Digital multiplexers, Adders and Subtractors, Binary comparators, Latches and Master-Slave Flip-Flops.
- Describe, Design and Analyze Synchronous and Asynchronous Sequential.
- Explain and design registers and Counters, A/D and D/A converters.

Module 1	**Hours**
Field Effect Transistors: Junction Field Effect Transistors, MOSFETs, Differences between JFETs and MOSFETs, Biasing MOSFETs, FET Applications, CMOS Devices. Wave-Shaping Circuits: Integrated Circuit (IC) Multi vibrators. Introduction to Operational Amplifier: Ideal v/s practical Opamp, Performance Parameters	10
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 Arrays, HDL Implementation of Data Processing Circuits. Arithmetic Building Blocks, Arithmetic Logic Unit Flip- Flops: RS Flip-Flops, Gated Flip-Flops, Edge-triggered RS FLIP-FLOP, Edge-triggered D FLIP-FLOPs, Edge- triggered JK FLIPFLOPs	10
Module III	
Linked Lists: Definition, Representation of linked lists in Memory, Memory allocation; Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists and header linked lists. Linked Stacks and Queues. Applications of Linked lists – Polynomials, Additional list operations-inverting singly linked list, concatenating singly linked list. Sparse matrix representation.	08
Module IV	
Flip- Flops: FLIP-FLOP Timing, JK Master-slave FLIP-FLOP, Switch Contact Bounce Circuits, Various Representation of FLIP-FLOPs, HDL Implementation of FLIP-FLOP. 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, Register implementation in HDL. Counters: Asynchronous Counters, Decoding Gates,3s	10
Module V	
Design of synchronous and asynchronous sequential circuits: model selection, state transition diagram, state synthesis table design equation and circuit diagram, implementation using read only memory. **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.	10

Course Outcomes (COs):

CO1	Design and analyse application of analog circuits using Field Effect Transistors (FETs) and Operational Amplifiers (Op-Amps).
CO2	Simplify digital circuits using Karnaugh Maps (K-Map) and Quine-McClusky methods
CO3	Design different data processing circuits and develop simple Hardware Description Language (HDL) programs.
CO4	Analyse and implement Flip-Flops, Registers, and Counters.
CO5	Develop synchronous and asynchronous sequential circuits, and Digital-to-Analog (D/A) & Analog-to-Digital (A/D) converters.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	-	-	-	-	-	-	1	1	2	-
CO2	3	3	2	-	-	-	-	-	-	-	-	1	-	1	-
CO3	3	1	2	-	-	-	-	-	-	-	-	1	-	3	-
CO4	3	3	2	-	-	-	-	-	-	-	-	1	1	3	-
CO5	3	3	2	-	-	-	-	-	-	-	-	1	1	1	-

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Anil K Maini, Varsha Agarwal: Electronic Devices and Circuits, Wiley, 2012.Donald P Leach, Albert Paul Malvino & Goutam Saha:
2. Digital Principles and Applications, 8th Edition, Tata McGraw Hill, 2015

Reference Books:

1. Stephen Brown, Zvonko Vranesic: Fundamentals of Digital Logic Design with VHDL, 2nd Edition,Tata McGraw Hill, 2005.
2. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.
3. M Morris Mano: Digital Logic and Computer Design, 10 th Edition, Pearson, 2008

Computer Organization and Architecture

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18CS34	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 04

Course Objectives: This course will enable students to

- How Computer Systems work & the basic principles
- Instruction Level Architecture and Instruction Execution.
- The current state of art in memory system design
- How I/O devices are accessed and its principles.
- To provide the knowledge on Instruction Level Parallelism.
- To impart the knowledge on micro programming.
- Understand Concepts of advanced pipelining techniques, Computer Arithmetic and parallel processing

Module 1	**Hours**
Functional blocks of a computer: Functional units, Basic operational concepts, Bus Structure, Software, and Performance. Signed number representation, character representation. Memory location and address, Instruction and sequencing, Basic IO operations, Addressing Modes, Additional Instructions: Shift and Rotate Instructions	10
Module II	
Basic Processing Unit: Single Bus Organization, Multiple Bus Organization, Hardwired and micro-programmed design approaches. **Input Output Organization**: Accessing I/O devices, Interrupts, DMA, Buses.	10
Module III	
The Memory System: Semiconductor RAM memories (SDRAM, ADRAM), Cache Memories, Performance Consideration. **Arithmetic:** Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication	10
Module IV	
Pipelining: Introduction, Major Hurdles of Pipelining, how is pipelining implemented? What makes pipeline hard to implement, Instruction Level Parallelism: Concepts and Challenges	10
Module V	
Memory Hierarchy: Introduction, Cache Performance, Six basic Cache Optimization, Virtual Memory, Memory Hierarchy Design: 10 Advanced optimizations of cache Performances	08

Course Outcomes (COs):

CO1	Identify basic structure of computer and its performance measures.
CO2	Demonstrate functioning of bus structure, processor, Input/output
CO3	Design and analyze simple arithmetic and logical units and memory
CO4	Analyze the implementation of Pipelining and parallel processor.
CO5	Understand basic structure of computer memory and its performance measures using cache.

CO-PO-PSO mapping:

	PO 1	PO 2	PO 3	PO 4	PO5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	2	1	3	-
CO2	2	3	2	-	-	-	-	-	-	-	-	2	1	3	-
CO3	2	2	3	-	-	-	-	-	-	-	-	2	1	3	-
CO4	2	3	2	-	-	-	-	-	-	-	-	2	1	3	-
CO5	2	3	3	-	-	-	-	-	-	-	-	2	1	3	-

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1.Carl Hamacher, Z. Vranesic & S.Zaky, “Computer Organization”,5th Edition , Tata McGraw-Hill Publishing Company Ltd. New Delhi, 2002.

2.John L. Hennessy and David A. Patterson, Computer Architecture: A quantitative approach, 5th edition,Morgan Kaufmann Elseveir, 2013

Reference Books:

1. Morris Mano, “Computer System Architecture”, PHI, 19862.William Stallings Computer Organization & Architecture, 7th Edition, PHI 2006.
2. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability, McGraw Hill Education 3/e. 2015.

Data Structures Lab

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18CSL35	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03

CREDITS – 01

Course Objectives: This course will enable students

- To design, develop, test and debug in C/C++ language considering appropriate data structure.
- Illustrate and implement data types such as stack, queue and linked list and apply them for the given problem.
- Illustrate and implement the trees and other data structures.

PART-A

Students are required to implement following programs using C/C++.

1. Implementation of stack ADT using arrays
2. Implementation of queue ADT using arrays
3. Implementation of List ADT
4. Implementation of Graph ADT using List
5. Implementation of tree ADT using List / Array

PART-B

Application of Stack

1. Implementation of Infix to Postfix conversion.
2. Implementation of postfix evaluation.

Application of Queue

3. Implementation of Priority queue program using array.
4. Implementation of multiple stacks and queues

Application of List

5. Implementation of sparse matrix multiplication.
6. Implementation of Linked Lists menu driven program (stack and queue)

Application of Graph & Tree

7. Implementation of construction of expression tree using postfix expression.
8. Implementation of various operations on tree like – copying tree, counting the number of nodes in the tree.
9. Implementation of Binary Heap program

Course Outcomes (COs):

CO1	Demonstrate theoretical concepts of Arrays, Queues, stack, Linked list, graphs & trees data structures through series of experiments.
CO2	Implement various data structures using C/C++
CO3	Debug syntactical errors, and troubleshoot the problems issues effectively
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized Data Structures laboratory report.

CO-PO-PSO mapping:

	PO 1	PO 2	PO 3	PO 4	PO5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	-	-	-	-	-	-	-	-	2	1	1	1
CO2	2	2	3	-	-	-	--	--	-	-	-	2	1	2	3
CO3	2	2	2	-	-	-	-	-	-	-	-	-	1	2	3
CO4	2	3	1	-	-	-	-	-	-	-	-	-	1	3	2
CO5	1	-	-	-	-	-	-	-	-	3	-	-	1	-	-

Conduct of Practical Examination:

- Experiment distribution

- For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
- For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

- Marks Distribution (Courseed to change in accoradance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks
- For laboratories having PART A and PART B

Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

iv. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

Electronics Circuits & Logic Design Lab

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18CSL36	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03
CREDITS – 01			
Course Objectives: This course will enable students			

- Provide hands-on experience in designing, analyzing, and implementing electronic circuits and logic gates,
- Fostering practical skills in areas like amplifier circuits, op-amp applications, and digital logic

PART A

1. A. Design and construct a Schmitt trigger using Op-Amp for given UTP and LTP values and demonstrate its working.

   B. Design and implement a Schmitt trigger using Op-Amp using a simulation package for two sets of UTP and LTP values and demonstrate its working.

2. A. Design and construct a rectangular waveform generator (Op-Amp relaxation oscillator) for given frequency and demonstrate its working.

   B. Design and implement a rectangular waveform generator (Op-Amp relaxation oscillator) using a simulation package and demonstrate the change in frequency when all resistor values are doubled.

3. Design and implement an A stable multi vibrator circuit using 555 timer for a given frequency and duty cycle.

PART B

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

2. Given a 4-variable logic expression, simplify it using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC.
3. Design and develop the Verilog /VHDL code for an 8:1 multiplexer. Simulate and verify it's working.
4. Design and implement code converter
   I) Binary to Gray
   II) Gray to Binary Code using basic gates.

5. Design and verify the Truth Table of 3-bit Parity Generator and 4-bit Parity Checker using basic Logic Gates with an even parity bit.
6. a) Realize a J-K Master / Slave Flip-Flop using NAND gates and verify its truth table.
   b) Design and develop the Verilog / VHDL code for D Flip-Flop with positive edge triggering.

Course Outcomes (COs):

CO1	Demonstrate theoretical concepts in electronics circuits and logic design through practical experiments.
CO2	Create a logic circuit using appropriate devices.
CO3	Identify, debug, and resolve issues effectively.
CO4	Analyze data from experiments and interpret the results accurately.
CO5	Prepare a well-organized laboratory report detailing experimental procedures, results

CO-PO-PSO mapping:

	PO 1	PO 2	PO 3	PO 4	PO5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	-	-	-	-	-	-	-	1	3	3
CO2	2	2	3	-	2	-	-	-	-	-	-	-	1	2	3
CO3	2	1	-	-	2	-	-	-	-	-	-	-	-	2	3
CO4	2	3	2	-	-	-	-	-	-	-	-	1	-	-	2
CO5	2	1	-	-	-	-	-	-	-	-	-	1	-	1	2

Conduct of Practical Examination:

- Experiment distribution

● For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.

● For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

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

- Marks Distribution (Courseed to change in accoradance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks
- For laboratories having PART A and PART B

Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

vi. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

Unix Shell Programming Lab

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18CSL37	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03
CREDITS – 01			
Course Objectives: This course will enable students			

To Study of UNIX basic Commands
To introduce Basic Unix general purpose Commands.
To write shell scripts to solve problems

PART A

1. Study of UNIX basic commands:
   cal, date, echo, printf, bc, script, mailx, passwd, who, uname, tty, stty, pwd, cd, mkdir, rmdir, ls, cat, cp, rm, mv, more, file, wc, od, cmp,comm, diff, chmod.
2. Study of vi editor.
3. Write a script to study if...else, if and case statements.
4. Write a script to study for, while and until.
5. Study the Filters for stream handling features of the shell for input and output.
   E.g. pr, head, tail, cut, paste, sort, nl, uniq, tr.

PART B

1.

a) Write a Shell program to count number of user's login and print first login user information

b) Write Shell Script to read user name and find whether the user is currently working in the system or not.

2.

a) Write shell script for-
(i) Showing the count of users logged in.
(ii) Printing Column list of files in your home directory.
(iii) Listing your job with below normal priority.
(IV) Continue running your job after logging out.

b) Write a shell script to create a file. Follow the instructions
(i) Input a page profile to yourself, copy it into other existing file;
(ii) Start printing file at certain line.
(iii) Print all the difference between two file, copy the two files.
(iv) Print lines matching certain word pattern.

3.

a) Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.

b) Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.

4.

a) Write a shell script that receives any number of file names as arguments checks if every argument is a file or directory, when it is a file, report no of lines in it.

b) Write a shell script that accepts a list of file names as its arguments, count and reports the

Course Outcomes (COs):

CO1	Demonstrate theoretical concepts of UNIX SHELL PROGRAMMING through series of experiments.
CO2	Develop a program using software tools
CO3	Debug and troubleshoot software issues effectively
CO4	Analyse the data and interpret the results
CO5	Prepare a well-organized laboratory report

CO-PO-PSO mapping:

	PO 1	PO 2	PO 3	PO 4	PO5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	1	-	3	-	1	1	2	1	1	1	2	3	-
CO2	3	3	3	-	3	-	1	1	2	1	1	1	2	3	-
CO3	3	2	3	-	3	-	1	1	3	1	2	1	2	2	-
CO4	2	3	1	-	3	-	1	1	2	1	1	1	2	1	-
CO5	2	2	1	-	2	-	-	1	2	1	-	1	2	1	-

Conduct of Practical Examination:

- Experiment distribution

- For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
- For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

- Marks Distribution (Courseed to change in accoradance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks
- For laboratories having PART A and PART B

Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

viii. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

PROJECT III

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18CSL38	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03

CREDITS – 01

Course Objectives: This course will enable students

- Identify real-world problems across programming, databases, and networking domains and understand their business and technical implications.
- Apply systematic methodologies to design, implement, and optimize solutions.
- Resolve technical challenges through debugging, research, and collaboration.
- Take responsibility for specific roles in a team and collaborate effectively to achieve project goals.
- Present project progress and findings clearly and confidently to both technical and non-technical audiences.
- Document the entire project in a structured, professional laboratory report.

Project Guidelines:

- Project work shall preferably be batch wise.
- Evaluation is based on concept clarity, system design, implementation, testing, presentation, and documentation quality, with a focus on proper coding standards, teamwork, and effective communication.
- Viva-voce examination in project work shall be conducted batch-wise.
- Minimum requirement of CIE marks for Project work shall be 50% of the maximum marks.
- Students failing to secure a minimum of 50% of the CIE marks in Project work shall not be eligible for the SEE Project examination.
- For a pass in a Project/Viva-voce examination, a student shall secure a minimum of 40% of the maximum marks prescribed.

Course Outcomes (Cos):

CO1	Identify the topic from various domains (example programming databases, networking) to real world problems.
CO2	Develop methodology for the problem.
CO3	Resolve issues that arise during the project .
CO4	Learn to assign and accept roles and responsibilities within a team and write a good technical reports.
CO5	Exhibit skills in presenting their project findings & progress orally

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	2	2	-	1	1	1	2	2	1	2	2	3	2
CO2	2	2	2	2	2	1	2	1	2	2	1	1	2	2	2
CO3	2	2	2	2	3	1	1	1	2	2	1	1	2	3	2
C04	-	-	-	-	-	1	-	2	2	3	1	1	1	1	1
CO5	-	-	-	-	1	1	-	2	2	3	1	1	1	1	1

KANNADA KALI-3

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18KANKK310	**CIE Marks**	50
Number of Lecture Hours/Week	01	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	02

CREDITS - 01

Course Objectives: This course will enable students to

1) C£ Àå¨sÁ¶PÀ «zÁåyðUÀ½UÉ PÀ£ÀßqÀ ªÀiÁvÀ£ÁqÀÄªÀÅzÀÄ §gÉAiÀÄÄªÀ PË±À®å PÀ°¸ÀÄªÀÅzÀÄ.
2) PÀ£ÀßqÀ ¨sÁµÁ eÁÕ£ÀzÀ CjªÀÅ ªÀÄÆr¸ÀÄªÀÅzÀÄ.
3) PÀ£ÀßqÀ §gÀªÀt ÂUÉ PÀÄjvÀÄ w¼ÀÄªÀ½PÉ ªÀÄÆr¸ÀÄªÀÅzÀÄ.
4) PÀ£ÀßqÀ £Áq ÀÄ £ÀÄr, ¸ÀA¸ ÀÌøwAiÀÄ §UÉÎ w½¸ÀÄªÀÅzÀÄ.
5) PÀ£ÀßqÀ ¨sÁµÁ ¥ÉæÃªÀÄªÀ£ ÀÄß ¨É¼É¸ÀÄªÀÅªÀÅzÀÄ.

MODULE I

Lesson 1. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

Lesson 2. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

MODULE II

Lesson 3. Conversation 1, Conversation 2, Conversation 3,ocabulary, Exercises.

Lesson 4. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

MODULE III

Lesson 5. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

Lesson 6. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

MODULE IV

Lesson 7. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

Lesson8. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

MODULE V

Lesson 9. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

Lesson 10. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

Course Outcomes (COs):

CO1	To understand the necessity of local language for comfortable life.
CO2	To speak, read write Kannada language as per requirement.
CO3	To communicate [converse] in Kannada language in their daily life with Kannada speakers.
CO4	To listen and understand the Kannada language properly.
CO5	To speak in polite conversation.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	-	-	-	-	-	1	-	-	-	3	-	3	-	-	3
CO2	-	-	-	-	-	1	-	1	-	3	-	3	-	-	3
CO3	-	-	-	-	-	1	-	1	-	3	-	-	-	-	3
C04	-	-	-	-	-	1	-	-	-	3	-	-	-	-	3
CO5	-	-	-	-	-	1	-	2	-	2	-	-	-	-	3

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1.Carl Hamacher, Z. Vranesic & S.Zaky, "Computer Organization",5th Edition , Tata McGraw-Hill Publishing Company Ltd. New Delhi, 2002.

2.John L. Hennessy and David A. Patterson, Computer Architecture: A quantitative approach, 5th edition,Morgan Kaufmann Elseveir, 2013

Reference Books:

3. Morris Mano, "Computer System Architecture", PHI, 19862.William Stallings Computer Organization & Architecture, 7th Edition, PHI 2006.

4. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability, McGraw Hill Education 3/e. 2015.

AYDA KATHEGALU (DAiÀÄÝ PÀvÉUÀ¼ÀÄ)

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	20KANAK310	**CIE Marks**	50
Number of Lecture Hours/Week	01	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	02

CREDITS - 01

Course Objectives: This course will enable students to

1) PÀ£ÀßqÀ ¨sÁµÁ eÁÕ£ÀzÀ CjªÀÅ ªÀÄÆr¸ÀÄªÀÅzÀÄ.
2) PÀ£ÀßqÀ §gÀªÀtÂUÉ PÀÄjvÀÄ w¼ÀÄªÀ½PÉ ªÀÄÆr¸ÀÄªÀÅzÀÄ.
3) PÀ£ÀßqÀ £ÁqÀÄ £ÀÄr, ¸ÀA¸ÀÌøwAiÀÄ §UÉÎ w½¸ÀÄªÀÅzÀÄ.
4) PÀ£ÀßqÀ ¨sÁµÁ ¥ÉæÃªÀÄªÀ£ÀÄß ¨É¼É¸ÀÄªÀÅªÀÅzÀÄ.

MODULE I

1) ªÉÆ¸Àj£À ªÀÄAUÀªÀÄä- ªÀiÁ¹Û ªÉAPÀmÉÃ±À LAiÀÄåAUÁgÀ (²æÃ¤ªÁ¸À)
2) PÉÆ£ÉAiÀÄVgÁQ - ¤gÀAd£À

MODULE II

3) zÁj – avÀæ±ÉÃRgÀ PÀAp
4) ªÀiÁV – PÉÃ±ÀªÀ ªÀÄ¼ÀV

MODULE III

5) PÁqÀÄ - ¹zÀÝgÁªÀÄ ºÉÆ£ÀÌ¯ï
6) D¸ÉAiÉÄA§ vÀxÁUÀvÀ£À ªÉÊj – azÁ£ÀAzÀ ¸Á°

MODULE IV

7) vÀ§â°UÀ¼ÀÄ – gÁWÀªÉÃAzÀæ SÁ¸À¤Ã¸À
8) ¤ªÀÈvÀÛgÀÄ - ¦. ®APÉÃ±À

MODULE V

9) C§ZÀÆj£À ¥ÉÆÃ¸ÁÖ¦üÃ¸ÀÄ– PÉ.¦ ¥ÀÆtðZÀAzÀæ vÉÃd¹é
10) ºÀAV£ÀgÀªÀÄ£ÉAiÀÄ ºÉÆgÀUÉ–gÁd±ÉÃRgÀ ¤ÃgÀªÀiÁ¤é

Course Outcomes (Cos):

CO1	PÀ£ÀßqÀ ¸Á»vÀå §UÉÎ CjvÀÄPÉÆ¼ÀÄîvÁÛgÉ.
CO2	PÀ£ÀßqÀ ¨sÁµÁeÁÕ£ÀzÀ ªÀÄºÀvÀéªÀ£ÀÄß w½zÀÄPÉÆ¼ÀÄîvÁÛgÉ.
CO3	¨sÁµÁ©üªÀiÁ£ÀªÀ£ÀÄß ¨É¼É¹PÉÆ¼ÀÄîvÁÛgÉ.
CO4	PÀ£ÀßqÀ ¸Á»vÀå PÀÈwUÀ¼À §UÉÎ D¸ÀQÛ ªÀÄÆqÀÄvÀÛzÉ.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1						1		1							3
CO2						1				3					3
CO3						1						3			3
C04						1		2							3

ENGINEERING MATHEMATICS-IV

(Common to all branches)

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019-2020)

SEMESTER-IV

Course Code:	**18MAT41**	**CIE Marks:**	**50**
Contact Hours/ Week:	**04**	**SEE Marks:**	**50**
Total Hours:	**48**	**Exam Hours:**	**03**

CREDITS - 04

Course Objectives: This course will enable students to

- Learn Fourier series and Fourier transforms.
- Conversant with numerical methods to solve ordinary differential equations, complex analysis, joint probability distribution and stochastic processes arising in science and engineering.

Module I	Hours
Fourier Series: Periodic functions, Dirichlet's condition, Fourier Series of periodic function with period 2π and with arbitrary period 2c. Fourier series of even and odd functions Half range Fourier Series, practical harmonic analysis(5 Assignment Problem)..	**10**
Module II	
Fourier Transforms: Infinite Fourier transforms, Fourier sine and cosine transforms. Inverse Fourier-transform (5 Assignment Problem). **Complex line Integrals:** Cauchy's Integration theorem, Cauchy integral formula, Laurent's Series, types of singularities. Residue, Poles, Cauchy's Residue theorem (without proof) and Problems. **Transformations:** Bilinear transformations and problems.	**08**
Module III	
Numerical Methods: Numerical solution of ordinary differential equations of first order and first degree, Taylor's series method, modified Euler's-method Runge Kutta method of fourth order. Milne's and Adams- Bashforth predictor and corrector methods (No derivations of formulae). (5 Assignment Problem).	**10**
Module IV	
Numerical Methods: Numerical solution of second order ordinary differential equations, Runge- Kutta Method and Milne's Method, Numerical solution of P.D.E: Numerical solution of heat equation, wave equation, problems. (5 Assignment Problem	**10**
Module V	
Joint probability distribution: Joint Probability distribution for two discrete random variables, expectation, covariance, correlation coefficient **Stochastic process:** Stochastic processes, probability vector, stochastic matrices, fixed points, regular stochastic matrices, Markov chains, higher transition probability- simple problems.(5 Assignment Problem).	**10**

Course Outcomes (COs):

CO1	Understanding the Periodic function and Fourier series expansion of different functions and its application to analyze circuits
CO2	Apply the knowledge of Fourier transform and Understand the complex potentials in different engineering fields
CO3	Solving the first order first degree ordinary differential equations arising in flow problems by numerical methods.
CO4	Make the use of second order ordinary and partial differential equations arising in heat and wave equations by numerical methods.
CO5	Learn to solve the problems on Joint probability distribution and to know the concept of stochastic processes and Markov's chains in discrete time.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-
CO2	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-
CO3	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-
CO4	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-
CO5	3	2	-	-	-	-	-	-	-	-	-	1	3	-	-

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1.B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, 43rd Ed., 2015.
2.E. Kreyszig:Advanced Engineering Mathematics, John Wiley & Sons,10th Ed., 2015.

Reference Books:

1. N.P.Bali and Manish Goyal: A Text Book of Engineering Mathematics, Laxmi Publishers , 7th Ed., 2010.
2. B.V.Ramana: "Higher Engineering Mathematics" Tata McGraw-Hill, 2006.

DESIGN AND ANALYSIS OF ALGORITHMS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – IV

Subject Code	18CS42	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course Objectives: This course will enable students to

Explain various computational problem-solving techniques.
Apply appropriate method to solve a given problem.

- Describe various methods of algorithm analysis.

Module I	Hours
Introduction: Notion of Algorithm, Review of Asymptotic Notations, Mathematical analysis of non-recursive and recursive Algorithms with Examples. Important Problem Types: Sorting, Searching, String processing, Graph Problems, Combinatorial Problems. Fundamental Data Structures: Stacks, Queues, Graphs, Trees, Sets and Dictionaries.	**10**
Module II	
Divide and Conquer: General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum, Merge sort, Quick sort , Advantages and Disadvantages of divide and conquer. Decrease and Conquer Approach: Topological Sorting.	**10**
Module III	
Greedy Method: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines. Minimum cost spanning trees: Prim's Algorithm, Kruskal's Algorithm. Single source shortest paths: Dijkstra's Algorithm . Optimal Tree problem: Huffman Trees and Codes . Transform and Conquer Approach: Heaps and Heap Sort.	**08**
Module IV	
Dynamic Programming: General method with Examples. Transitive Closure: Warshall's Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Knapsack problem,Bellman-Ford Algorithm, Travelling Sales Person problem.	**10**
Module V	
Backtracking: General method , N-Queens problem , Sum of subsets problem , Graph coloring , Hamiltonian cycles . Branch and Bound: Assignment Problem, Travelling Sales Person problem , 0/1 Knapsack problem : LC Branch and Bound solution , FIFO Branch and Bound solution.	**10**

Course Outcomes (COs):

CO1	Identify various time and space complexities of various algorithms for well known problems like searching, sorting etc.
CO2	Estimate the computational complexity of different algorithms using divide and conquer, decrease and conquer design paradigm.
CO3	Apply greedy techniques for solving the given real world problem.
CO4	Apply dynamic programming concept to solve various problem.
CO5	Implement the programs by using backtracking and branch and bound and analyze the complexities.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	1	-	-	-	-	-	-	-	-	-	2	3	1
CO2	2	3	2	-	-	-	-	-	-	-	-	-	2	3	1
CO3	2	3	1	1	1	-	-	-	-	-	-	-	2	3	1
CO4	2	3	1	-	1	-	-	-	-	-	-	-	2	3	1
CO5	2	3	1	-	1	-	-	-	-	-	-	-	2	3	1

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2nd Edition, 2009. Pearson.
2. Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.
2. Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education

MICROPROCESSOR & MICROCONTROLLER

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – IV

Subject Code	18CS43	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course Objectives: This course will enable students to

To impart basic understanding of the internal organization of 8086/88 Microprocessor.
To introduce the concepts of interfacing microprocessors with external devices.

- To develop Assembly language programming skills.

Module I	Hours
The x86 microprocessor: Brief history of the x86 family, Inside the 8088/86,Introduction to assembly programming, Introduction to Program Segments, The Stack, Flag register, x86 Addressing Modes. Assembly language programming: Directives & a Sample Program, Assemble, Link & Run a program, More Sample programs, Control Transfer Instructions, Data Types and Data Definition,	**10**
Module II	

x86: Instructions sets description, Arithmetic and logic instructions and programs: Unsigned Addition and Subtraction, Unsigned Multiplication and Division, Logic Instructions, BCD and ASCII conversion, Rotate Instructions. INT 21H and INT 10H Programming: Bios INT 10H Programming , DOS Interrupt 21H. 8088/86 Interrupts,x86 PC and Interrupt Assignment.	**10**
Module III	
Signed Numbers and Strings: Signed number Arithmetic Operations, String operations, Memory and Memory interfacing: Memory address decoding, data integrity in RAM and ROM, 16-bit memory interfacing. 8255 I/O programming: I/O addresses MAP of x86 PC's, programming and interfacing the 8255.	**08**
Module IV	
Microprocessors versus Microcontrollers, ARM Embedded Systems :The RISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software, ARM Processor Fundamentals : Registers , Current Program Status Register , Pipeline, Exceptions, Interrupts, and the Vector Table.	**10**
Module V	
Introduction to the ARM Instruction Set : Data Processing Instructions ,Branch Instructions, Software Interrupt Instructions, Program Status Register Instructions Coprocessor Instructions, Loading Constants.	**10**

Course Outcomes (COs):

CO1	Understand the architecture,features and basic instructions of 8086
CO2	Apply 8086 assembly language code to solve problems for arithmetic operations,code conversionand handle interrupts
CO3	Illustrate the design aspects of I/O and memory interfacing circuits.
CO4	Understand the architecture and features of ARM Embedded systems
CO5	Design and develop assembly language programs for ARM processor.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	1	1	-	-	-	-	-	-	-	-	1	1	2	1
CO2	2	1	1	-	-	-	-	-	-	-	-	2	1	2	-
CO3	2	1	1	-	-	-	-	-	-	-	-	2	-	2	-
CO4	1	1	1	-	-	-	-	-	-	-	-	1	-	2	1
CO5	2	1	1	-	-	-	-	-	-	-	-	2	-	2	1

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Danny Causey, The x86 PC Assembly Language Design and Interfacing, 5th Edition, Pearson, 2013.
2. ARM system developers guide, Andrew N Sloss, Dominic Symes and Chris Wright, Elsevier, Morgan Kaufman publishers, 2008.

Reference Books:

1. Douglas V. Hall: Microprocessors and Interfacing, Revised 2nd Edition, TMH, 2006.
2. Ayala : The 8086 Microprocessor: programming and interfacing - 1st edition, Cengage Learning
3. The Definitive Guide to the ARM Cortex-M3, by Joseph Yiu, 2nd Edition , Newnes, 2009
4. The Insider's Guide to the ARM7 based microcontrollers, Hitex Ltd.,1st edition, 2005
5. ARM System-on-Chip Architecture, Steve Furber, Second Edition, Pearson, 2015
6. Architecture, Programming and Interfacing of Low power Processors- ARM7, Cortex-M and MSP430, Lyla B Das Cengage Learning, 1st Edition

JAVA PROGRAMMING

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – IV

Subject Code	18CS44	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04	
Course Objectives: This course will enable students to	
- Learn fundamental features of object oriented language and JAVA - Set up Java JDK environment to create, debug and run simple Java programs. - Learn object oriented concepts using programming examples. - Study the concepts of importing of packages and exception handling mechanism. - Discuss the String Handling examples with Object Oriented concepts.	
Module I	**Hours**
An Overview of Java: Object-Oriented Programming, A First Simple Program, A Second Short Program, Two Control Statements, Using Blocks of Code, Lexical Issues, The Java Class Libraries, Data Types, Variables, and Arrays: Java Is a Strongly Typed Language, The Primitive Types, Integers, Floating-Point Types, Characters, Booleans, A Closer Look at Literals, Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays, A Few Words About Strings	**10**
Module II	
Operators: Arithmetic Operators, The Bitwise Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses, Control Statements: Java's Selection Statements, Iteration Statements, Jump Statements. Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection, The finalize( ) Method, A Stack Class.	**10**
Module III	
A Closer Look at Methods and Classes: Overloading Methods, Using Objects as Parameters, A Closer Look at Argument Passing, Returning Objects, Recursion, Introducing Access Control, Understanding static, Introducing final, Arrays Revisited, Inheritance: Inheritance, Using super, Creating a Multilevel Hierarchy, When Constructors Are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, The Object Class.	**10**
Module IV	
Packages and Interfaces: Packages, Access Protection, Importing Packages, Interfaces, Exception Handling: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions, Using Exceptions.	**08**
Module V	
The Applet Class: Introduction, Two types of Applets; Applet basics; Applet Architecture; An Applet skeleton; Simple Applet display methods; Requesting repainting; Using the Status Window; The HTML APPLET tag; Passing parameters to Applets; getDocumentbase() and getCodebase(); ApletContext and showDocument(); The AudioClip Interface; The AppletStub Interface; Output to the Console. String Handling: The String Constructors, String Length, Special String Operations, Character Extraction, String Comparison, Searching Strings, Modifying a String, Data Conversion Using valueOf( ), Changing the Case of Characters Within a String , Additional String Methods, StringBuffer, StringBuilder.	**10**

Course Outcomes (COs):

CO1	Understand the basics of object-oriented programming using C++ and JAVA.
CO2	Apply the concept of classes, Java, JDK Components and develop Simple Java Programs.
CO3	Develop Simple Java Programs using inheritance and Exception handling.

CO4	Develop Multi-threading Programming and Interfaces.
CO5	Develop GUI applications using Swing components and Event handling programs.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	-	3	2	1
CO2	2	3	2	-	-	-	-	-	-	-	-	-	3	2	1
CO3	2	2	3	-	-	-	-	-	-	-	-	-	3	2	2
CO4	2	2	3	-	-	-	-	-	-	-	-	-	3	2	2
CO5	2	2	2	2	-	-	-	-	-	-	-	-	3	3	2

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1 . Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007. (Chapters 2, 3, 4, 5, 6,7, 8, 9,10, 12,13,15)

Reference Books:

1. Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, Pearson Education,2008, ISBN:9788131720806.

2. Rajkumar Buyya,S Thamarasiselvi, xingchenchu, Object oriented Programming with java, Tata McGraw Hill education private limited.

3. E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies

MICROPROCESSOR AND MICROCONTROLLER LAB

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – IV

Subject Code	18CSL45	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03

CREDITS – 01

Course Objectives: This course will enable students to

Demonstration and Explanation of hardware components ,8086 architecture, pin diagram

- Develop and execute the following programs using 8086 Assembly Language. Any suitable assembler like MASM/TASM/8086 kit or any equivalent software may be used.

Laboratory Session-1: Write-up on Microprocessors, 8086 Functional block diagram, Pin diagram and description. The same information is also taught in theory class; this helps the students to understand better

Laboratory Session-2: Write-up on Instruction group, Timing diagrams, etc. The same information is also taught in theory class; this helps the students to understand better.

Note: These TWO Laboratory sessions are used to fill the gap between theory classes and practical sessions. Both sessions are to be evaluated for 20 marks as lab experiments.

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. 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.

4. Develop an assembly language program to reverse a given string and verify whether it is a palindrome or not. Display the appropriate message.

5. Design an assembly language program to compute nCr using recursive procedure. Assume that ‘n’ and ‘r’ are non-negative integers.

Design an assembly language program to create a file and delete an existing file.

7. To write and simulate C Program to ARM microprocessor using KEIL. (Demonstrate with the help of suitable program)

PART – B

1. 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.

Design and develop BCD Up-Down counter using Logic Controller Interface.

3. 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.

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

Course Outcomes (COs):

CO1	Demonstrate theoretical knowledge of microprocessors and microcontrollers with practical skills by conducting a series of hands-on experiments.
CO2	Develop a Program using MASM for x86 assembly language and ARM development tools for ARM architecture.
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	1	-	-	-	-	-	-	-	-	-	-	3	3
CO2	3	1	2	-	2	-	-	-	-	-	-	-	-	3	3
CO3	2	2	-	-	2	-	-	-	-	-	-	-	-	-	3
CO4	1	1	-	-	1	-	-	-	-	-	-	-	-	1	2
CO5	1	1	1	-	-	-	-	-	-	1	-	-	-	-	1

Conduct of Practical Examination:

- Experiment distribution

- For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
- For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

- Marks Distribution (Courseed to change in accoradance with university regulations)

- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks

- For laboratories having PART A and PART B

Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

x. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

JAVA PROGRAMMING LAB

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – IV

Subject Code	18CSL46	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03
CREDITS – 01			
Course Objectives: This course will enable students			

- Learn fundamental features of object oriented language and JAVA
- Set up Java JDK environment to create, debug and run simple Java programs. Learn object oriented concepts using programming examples

PART – A (Implement the following in JAVA)

1.a.Write a JAVA program to implement class mechanism. –Create a class, methods and invoke them inside main method.

b.Write a JAVA program to implement shift operators in JAVA

2.a.Write a JAVA program to implement constructor overloading.

b.Write a JAVA program to implement for-each loop to compute average of n natural numbers.

3.a.Write a JAVA program to implement multi level Inheritance.

b.Write a JAVA program for abstract class to find areas of different shapes.

4.a.Write a JAVA program that describes exception handling mechanism.

b.Write a JAVA program to implement break and continue statements.

5.a.Write a JAVA program using IO Streams.

b.Write a JAVA program using files.

PART – B (Implement the following in JAVA)

1.Write a JAVA program that creates threads by extending Thread class .First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds,(Repeat the same by implementing Runnable.

2.Write a JAVA program Producer Consumer Problem.

3.Write a JAVA program to create an applet and set its background color and foreground color displaying a message

4.Write a JAVA program to demonstrate key event handlers using delegation event model.

Course Outcomes (COs):

CO1	Demonstrate theoretical concepts of constructor, inheritance, threads and Exception Handling through series of experiments.
CO2	Develop a program using basic programming constructs and standard libraries.
CO3	Apply advanced debugging techniques and utilize integrated development environment (IDEs) to efficiently identify, diagnose, and resolve software issues in java applications.
CO4	Employ advanced data analysis technique and utilize java libraries to process, analyze and interpret data effectively.
CO5	Demonstrate theoretical concepts of constructor, inheritance, threads and Exception Handling through series of experiments.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	-	-	-	-	-	-	-	-	-	2	3	3
CO2	2	3	3	-	-	-	-	-	-	-	-	-	2	3	3
CO3	2	3	2	-	-	-	-	-	-	-	-	-	2	3	2
CO4	2	3	2	-	-	-	-	-	-	-	-	-	2	3	2
CO5	1	2	1	-	-	-	-	-	-	-	-	-	2	2	1

Conduct of Practical Examination:

- Experiment distribution

- For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
- For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

- Marks Distribution (Courseed to change in accoradance with university regulations)

- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks

- For laboratories having PART A and PART B

Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

xii. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

ALGORITHM ANALYSIS AND DESIGN LAB **[As per Choice Based Credit System (CBCS) scheme]** **(Effective from the academic year 2019-2020)** **SEMESTER – IV**			
Subject Code	18CSL47	**CIE Marks**	50

Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03

CREDITS – 01

Course Objectives: This course will enable students

Design and implement various algorithms in JAVA

- Employ various design strategies for problem solving.

Measure and compare the performance of different algorithms.

PART – A

1. Design a program to search a key element of n integers using binary search algorithm and compute time complexity
2. Design a program to Sort a given set of n integer elements using Quick Sort method and compute its time complexity.
3. Design a program to sort set of n integer elements using Merge Sort method and compute its time complexity.

Implement the 0/1 Knapsack problem using

(a) Dynamic Programming method.

(b) Greedy method.

5. Design a program to print all the node reachable from a given starting node in a given digraph using DFS method.

PART – B (Implement the following in JAVA)

Write a Program find shortest paths to other vertices using Dijkstra's algorithm.

2. (a) Write a program to find a Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm.

   (b) Write a program to find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.
3. Write a program to

   (a) Implement All-Pairs Shortest Paths problem using Floyd's algorithm

   (b) Implement transitive closure using warshall Algorithm.
4. Design and implement to find a subset of a given set.

Implement Travelling Salesman problem using Dynamic program.

Course Outcomes (COs):

CO1	Understanding of algorithmic design paradigms and the techniques used for analyzing their efficiency.
CO2	Implement programs using various design strategies
CO3	Debug and troubleshoot software issues effectively
CO4	Analyze the data and interpret the results
CO5	Prepare a well organized laboratory report

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO 3
CO1	3	2	3	-	-	-	-	1	-	-	-	-	1	3	3
CO2	2	2	3	-	1	-	-	1	-	-	-	-	1	3	3
CO3	1	3	-	-	-	-	-	1	-	-	-	-	1	1	3
CO4	1	2	2	-	-	2	-	-	-	-	-	-	1	1	1
CO5	1	-	-	-	-	-	-	-	-	3	-	-	-	1	-

Conduct of Practical Examination:

- Experiment distribution

- For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.

- For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
  - Marks Distribution (Courseed to change in accoradance with university regulations)
  - For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks
  - For laboratories having PART A and PART B

Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

xiv. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

PROJECT IV

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – IV

Subject Code	18CSL48	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03

CREDITS – 01

Course Objectives: This course will enable students

- Identify real-world problems across programming, databases, and networking domains and understand their business and technical implications.
- Apply systematic methodologies to design, implement, and optimize solutions.
- Resolve technical challenges through debugging, research, and collaboration.
- Take responsibility for specific roles in a team and collaborate effectively to achieve project goals.
- Present project progress and findings clearly and confidently to both technical and non-technical audiences.
- Document the entire project in a structured, professional laboratory report.

Course Outcomes (COs):

CO1	Identify the topic from various domains (example programming databases, networking) to real world problems.
CO2	Develop methodology for the problem.
CO3	Resolve issues that arise during the project .
CO4	Learn to assign and accept roles and responsibilities within a team and write a good technical reports.
CO5	Exhibit skills in presenting their project findings & progress orally

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	2	2	-	1	1	1	2	2	1	2	2	3	2
CO2	2	2	2	2	2	1	2	1	2	2	1	1	2	2	2
CO3	2	2	2	2	3	1	1	1	2	2	1	1	2	3	2
C04	-	-	-	-	-	1	-	2	2	3	1	1	1	1	1
CO5	-	-	-	-	1	1	-	2	2	3	1	1	1	1	1

Conduct of Practical Examination:

- Experiment distribution

- For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
- For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

- Marks Distribution (Courseed to change in accoradance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks
- For laboratories having PART A and PART B

Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

xvi. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

MAHADASOHI (ªÀÄºÁzÁ¸ÉÆÃ»UÀ¼ÀÄ)

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – IV

Subject Code	20KANMD410	**CIE Marks**	50
Number of Lecture Hours/Week	01	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	02

CREDITS - 01

Course Objectives: This course will enable students to

1) PÀ£ÀßqÀ ¨sÁµÁeÁÕ£ÀzÀCjªÀÅ ªÀÄÆr¸ÀÄªÀÅzÀÄ.
2) PÀ£ÀßqÀ §gÀªÀtÂUÉPÀÄjvÀÄ w¼ÀÄªÀ½PÉ ªÀÄÆr¸ÀÄªÀÅzÀÄ.
3) PÀ£ÀßqÀ £ÁqÀÄ £ÀÄr, ¸ÀA¸ÀÌøwAiÀÄ §UÉÎ w½¸ÀÄªÀÅzÀÄ.
4) PÀ£ÀßqÀ ¨sÁµÁ ¥ÉæÃªÀÄªÀ£ÀÄß ¨É¼É¸ÀÄªÀÅªÀÅzÀÄ.

MODULE I

1)±ÀgÀt§¸ÀªÉÃ±ÀéªÀgÀgÀ §zÀÄPÀÄ (CgÀ¼ÀUÀÄÄArUÉAiÀÄ°è)

2)¸ÁzsÀ£ÁPÉëÃvÀæ (OgÁzÀ,¥ÀvÁð¨Ázï ªÀÄvÀÄÛ PÀ®§ÄgÀVAiÀÄ°è)

MODULE II

3)zÉÆqÀØ¥ÀàC¥Àà ºÁUÀÆ ±ÀgÀt§¸ÀªÀ¥ÀàCªÀgÀ ¸ÀA§AzsÀUÀ¼ÀÄ (1 jAzÀ6£ÉÃ ¦ÃoÁ¢ü¥ÀwUÀ¼ÀÄ)

4)ªÀÄgÀÄ¼À ±ÀgÀt§¸À¥Àà (zÉÃªÁ®AiÀÄ ¤ªÀiÁðt, zÁ¸ÉÆÃºÀ ªÀÄºÁªÀÄ£ÉAiÀÄ ¨É¼ÀªÀtÂUÉ)

MODULE III

5)¥ÀÆdåzÉÆqÀØ¥ÀàC¥Àà (zsÁ«ÄðPÀ ¸ÁzsÀ£É)

6)±ÉÊPÀëtÂPÀ ¸ÁzsÀ£ÉUÀ¼ÀÄ

MODULE IV

7) ¥ÀÆdåqÁ. ±ÀgÀt§¸ÀªÀ¥ÀàC¥Àà (¸ÁªÀiÁfPÀ PÉÆqÀÄUÉUÀ¼ÀÄ)

8) ±ÉÊPÀëtÂPÀ PÉÆqÀÄUÉUÀ¼ÀÄ

MODULE V

9) ªÀÄºÁªÀÄ£ÉAiÀÄ ªÀÄºÁ ªÀiÁvÉAiÀÄgÀÄ ªÉÆzÀ® £Á®ÄÌ ¥ÀÄtå¹ÛçÃAiÀÄgÀÄ

10) LzÀ£ÉAiÀÄ ¦ÃoÁ¢ü¥ÀwU½AzÀ 8£ÉÃ ¦ÃoÁ¢ü¥ÀwUÀ¼À ¥ÀÄtå¹ÛçÃAiÀÄgÀÄ

Course Outcomes (Cos):

CO1	PÀ£ÀßqÀ ¸Á»vÀå §UÉÎ CjvÀÄPÉÆ¼ÀÄîvÁÛgÉ.
CO2	PÀ£ÀßqÀ ¨sÁµÁeÁÕ£ÀzÀ ªÀÄºÀvÀéªÀ£ÀÄß w½zÀÄPÉÆ¼ÀÄîvÁÛgÉ
CO3	¨sÁµÁ©üªÀiÁ£ÀªÀ£ÀÄß ¨É¼É¹PÉÆ¼ÀÄîvÁÛgÉ.
C04	PÀ£ÀßqÀ ¸Á»vÀå PÀÈwUÀ¼À §UÉÎ D¸ÀQÛ ªÀÄÆqÀÄvÀÛzÉ

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1						1		1							3
CO2						1				3					3
CO3						1						3			3
C04						1		2							3

KANNADA KALI-4

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18KANKK410	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	30	**Exam Hours**	03

CREDITS - 01

Course Objectives: This course will enable students to

- C£Àå¨sÁ¶PÀ «zÁåyðUÀ½UÉ PÀ£ÀßqÀ ªÀiÁvÀ£ÁqÀÄªÀÅzÀÄ
- §gÉAiÀÄÄÄªÀPË±À®å PÀ°¸ÀÄªÀÅzÀÄ.
- PÀ£ÀßqÀ ¨sÁµÁeÁÕ£ÀzÀCjªÀÅ ªÀÄÆr¸ÀÄªÀÅzÀÄ.
- PÀ£ÀßqÀ §gÀªÀtÂUÉPÀÄjvÀÄ
- w¼ÀÄªÀ½PÉ ªÀÄÆr¸ÀÄªÀÅzÀÄ.
- PÀ£ÀßqÀ £ÁqÀÄ £ÀÄr, ¸ÀA¸ÀÌøwAiÀÄ §UÉÎ w½¸ÀÄªÀÅzÀÄ.
- PÀ£ÀßqÀ ¨sÁµÁ ¥ÉæÃªÀÄªÀ£ÀÄß ¨É¼É¸ÀÄªÀÅªÀÅzÀÄ.

MODULE I

Lesson 1. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

Lesson 2. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

MODULE II

Lesson 3. Conversation 1, Conversation 2, Conversation 3,ocabulary, Exercises.

Lesson 4. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

MODULE III

Lesson 5. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises. **Lesson 6.**Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.
MODULE IV
Lesson 7. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises. **Lesson8.**Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.
MODULE V
Lesson 9. Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises. **Lesson 10.**Conversation 1, Conversation 2, Conversation 3,Vocabulary, Exercises.

Course Outcomes (COs):

CO1	To understand the necessity of local language for comfortable life.
CO2	To speak, read write Kannada language as per requirement.
CO3	To communicate [converse] in Kannada language in their daily life with Kannada speakers.
CO4	To listen and understand the Kannada language properly.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1						1		1							3
CO2						1				3					3
CO3						1						3			3
C04						1		2							3