P D A College of Engineering

B.E. Information Science & Engineering

Scheme of Teaching and Examinations 2022

Outcome Based Education(OBE)and Choice Based Credit System(CBCS) - (Effective from the academic year 2023-24)

IIISEMESTER

Sl.No	**Course**	**Course Code**	**CourseTitle**	**TeachingDepartment (TD)and QuestionPaper SettingBoard(PSB)**	**TeachingHours/Week**				**Examination**				**Credits**
					TheoryLectur	**Tutorial**	**Practical/Drawing**	**SDA**	**Duration inhours**	**CIEMarks**	**SEEMarks**	**TotalMarks**	
					L	**T**	**P**	**S**					
1	PCC/BSC	22MATS31	Engineering Mathematics-III For Computer Science And Engineering Stream	TD-RespectiveDept. PSB- RespectiveDept.	2	2	0		03	50	50	100	3
2	IPCC	22IS32	Data Structure Using C	TD-RespectiveDept. PSB- RespectiveDept.	3	0	2		03	50	50	100	4
3	IPCC	22IS33	Object oriented Prog Using C++	TD-RespectiveDept. PSB- RespectiveDept	3	0	2		03	50	50	100	4
4	PCC	22IS34	Digital System Design & Organization	TD-RespectiveDept. PSB- RespectiveDept	3	0	0		03	50	50	100	3
5	PCCL	22ISL35	Digital System Design & Orgn. Lab	TD-RespectiveDept. PSB- RespectiveDept	0	0	2		03	50	50	100	1
6	ESC	22IS36A	Discrete Mathematical Structure and Graph Theory	TD:RespectiveDept. PSB:Respective Dept.	3	0	0		03	50	50	100	3
7	UHV	22UHV37	Social Connect and Responsibility	Any Department	0	0	2		02	50		50	1
8	AEC/SEC	22ISAE381	Ability Enhancement Data Visualization using Python		If the course Is a Theory					50	50	100	1
					0	0	2		02				
					If a course is a laboratory								
					0	0	2		03				
9	NCMC	22NS39	Mandatory Course	NSS Coordinator	0	0	2			50	---	50	0
		22PE39	Mandatory Course	Physical Education Director									
		22YO39	Mandatory Course	YogaTeacher									
Total										**450**	**350**	**800**	**20**

PCC: Professional Core Course, PCCL: Professional Core Course laboratory, UHV: Universal Human Value Course, MC: Mandatory Course (Non-credit), AEC: AbilityEnhancement Course, SEC:Skill Enhancement Course, L: Lecture, T: Tutorial, P: Practical S= SDA: Skill Development Activity, CIE: Continuous Internal Evaluation,
SEE:Semester End Evaluation.:This letter in the course code indicates common to all the stream of engineering.
ESC:EngineeringScienceCourse, ETC:Emerging Technology Course, PLC:Programming Language Course
Professional Core Course (IPCC): Refers to Professional Core Course Theory Integrated with practical's of the same course. Credit for IPCC can be 04 and its Teaching–Learning hours (L : T : P) can be considered as(3 : 0 : 2) or (2 : 2 : 2). The theory part of the IPCC shall be evaluated both by CIE and SEE. The practical part shall be evaluated by only CIE(noSEE).However, questions from the practical part of I PCC shall be included in the SEE question paper.Form the regulation governing the Degree of Bachelor of Engineering/Technology(B.E./B.Tech.)2022-23may please be referred.

National Service Scheme /Physical Education/Yoga:All students have to register for any one of the courses namely National Service Scheme (NSS), Physical Education(PE)(Sports and Athletics), andYoga(YOG) with the concerned coordinator of the course during the first week of III semesters.Activities shall be carried out between IIIsemester to the VI semester (for 4 semesters). Successful completion of the registered course and requisite CIE score is mandatory for the award of the degree.
Theeventsshallbeappropriatelyscheduledbythecollegesandthesameshallbe reflected in the calendar prepared for the NSS,PE,and Yoga activities. These courses shall Not be considered for vertical progression as well as for the calculation of SGPA and CGPA, but completion of the course is mandatory for the award of degree.

HKE Society’s

PDA COLLEGE OF ENGINEERING,KALABURAGI

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

SYLLABUS

B.E III Semester

ENGINEERING MATHEMATICS-III FOR COMPUTER SCIENCE AND ENGINEERING STREAM

Course Code	**22MATS31**	Credits:03
CIE:50	SEE:50	SEE:3hrs
Hours/Week:03		Total hours:42

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. Probability distribution of discrete and continuous random variables
2. Joint Probability distributions and discrete and continuous random variables
3. Analyse the sample data using Large sample test, t-distribution and chi-distribution

Modules	**Teaching Hours**
Module-I **Probability distributions**: Random variable (Discrete and continuous) probability density function, cumulative density function, binomial distribution, poison distribution, Normal distribution and problems	9 hours
Module-II **Joint Probability distributions**: Concept of joint probability distribution, discrete and continuous random variables independent random variables, problems on exception and variance.	8 hours
Module -III **Samping theory-I**: 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	9 hours

Module -IV **Sampling Theory-II:** 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.	8hours
Module-V .**Optimization techniques**: Linear Programming: Mathematical formulation of linear programming problem(LPP), Types of solutions, Graphical method, Basic feasible solution, canonical and standard forms and simplex method	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. Higher Engineering Mathematics by B.S.Grewal, Khanna publishers; 40th Edition.2007

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

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 byS.S.Sastry

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Solve problems using theoretical probability distributions
	CO2	Apply the concepts of joint probability to find covariance, correlation, independent variables
	CO3	Analyze the sample data using Large sample tests
	CO4	Analyze the sample data using t-distribution and chi-distribution
	CO5	Apply optimization techniques and LPP for real life problems

DATA STRUCTURES USING C		
Course Code	**22IS32**	Credits:04
CIE:50	SEE:50	SEE:3hrs
Hours/Week:04(T+L)		Total hours:(40+12) =52
Prerequisite: The Students should have the thorough knowledge of C fundamentals		
Course objectives: To enable the students to obtain the knowledge of Data Structures using 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, Stings		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, Doubly linked Lists. Trees: Introduction, Binary Trees, Binary Tree Traversals.		8 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 .	8 hours

List of Programs for Data Structures Using C Lab

1. Design, Develop and Implement a menu-driven Program in C for the following Array operations
   Creating an Array of N Integer Elements
   Display of Array Elements with Suitable Headings
   Inserting an Element (ELEM) at a given valid Position (POS)
   Deleting an Element at a given valid Position(POS)
   Exit.
   Support the program with functions for each of the above operations Implement structures using C programs
2. Write a C program to create a sequential file with at least five records. Each record having the structure show below:
   Name Marks1 Marks2 Marks3
   Non-Zero
   Positive 25-Character Positive Integer Positive Integer Positive
   Integer
   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.
3. Design a C program to show the usage of Dynamic memory allocation techniques
4. Design, Develop and Implement a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX).
   Push an Element on to Stack
   Pop an Element from Stack
   Demonstrate Overflow and Underflow situations on Stack
   Display the status of Stack
   Exit
5. Design, Develop and Implement a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesizedand free parenthesized expressions with the operators: +, -, *, /, %(Remainder), ^(Power) and alphanumeric operands.
6. .Design, Develop and Implement a Program in C for the following Stack Applications
   i)Evaluation of Suffix expression with single-digit operands and operators:+, -, *, /, %, ^
   ii)Solving Tower of Hanoi problem with n disks.
7. Design, Develop and Implement a menu driven Program in C for the following operations on QUEUE of Characters (Array Implementation of Queue with maximum size MAX)
   Insert an Element on to QUEUE
   Delete an Element from QUEUE
   Demonstrate Overflow and Underflow situations on QUEUE
   Display the status of QUEUE Exit

8. Design, Develop and Implement a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX)
   Insert an Element on to Circular QUEUE
   Delete an Element from Circular QUEUE
   Demonstrate Overflow and Underflow situations on Circular QUEUE
   Display the status of Circular QUEUE
   Exit

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 (interger), student name(character string) and semester(integer). The operations to be supposed are:
   The insertion operation
   i)At the front of a list
   ii) At the back of 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 operation should be demonstrated.
   c)Displaying all the nodes in the list

10. Design, Develop and Implement a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo
    Create a DLL of N Employees Data by using end insertion.
    Display the status of DLL and count the number of nodes in it
    Perform Insertion and Deletion at End of DLL
    Perform Insertion and Deletion at Front of DLL
    Demonstrate how this DLL can be used as Double Ended Queue Exit
11. Write a C program i)To construct a binary search three of integers.

    ii)To traverse the tree using all the methods i.e, inorder, preorder and postorder.

12. Implement hashing technique to map a given key K to the address space L.Resolve the collision (if any) using linear probing. .

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	Implement arrays and structures using C programming.
	CO2	Implement different stack operations and recursive programs.
	CO3	Implement queue and linked list operations
	CO4	Implement tree traversal techniques using C Programming
	CO5	Implement hashing techniques using C programming.

OBJECT ORIENTED PROGRAMMING WITH C++ (IPCC)		
CourseCode	**22IS33**	Credits:04
CIE:50	SEE:50	SEE:3hrs
Hours/Week:04(T+L)		Total hours:(40+12)=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.		8 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, Destructors		8 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	8 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.	8 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.	8 hours

List of Programs for Object oriented programming with C++ Lab

1. Create a simple class STUDENT containing data members roll no, name age & display the contents using setdata() 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 aC++ 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.

6. Write a C++program to create a class called STACK using an array of integers. i)Implement the following operations by overloading + &-.s1=s1+element;where s1is an object of the class STACK and element is an integer to be pushed on to top of the stack.s1=s1- ; where s1 is an object of the class STACK and – operator pops the element. Handle the STACK Empty and STACK Full conditions. Also display the contents of the stack after each operation by overloading the operator<<.

7. Write a C++ program to create a class NAME and implement the following operations .Display the result after every operation by overloading the<<.
   i) NAMEfirstname="Herbert"

   ii) NAMElastname="Schield"

   iii) NAME fullname=firstname+lastname(Usecopyconstructor)

8. 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 m2to be added and subtracted. Perform the addition and subtraction
   By overloading the operators + and –respectively. Display the results(sum matrix m3and difference matrix m4) by overloading the operator<<.
   if(m1 = =m2)
   {
   m3=m1+m2;m4 = m1– m2;
   }
   Else
   Display error.

9. 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)–wheres1 is an integer (realpart)and s2 is a complex number.

ii. ADD(s1,s2)–wheres1and s2arecomplex numbers.

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

11. 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.

12. 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).

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 Senior_citizen = ¼ of adult ticket charge).

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

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++.

DIGITAL SYSTEM DESIGN AND ORGANIZATION

Course Code	**22IS34**	Credits:03
CIE:50	SEE:50	SEE:3hrs
Hours/Week:03		Total hours:42

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

Course objectives: To enable the students to obtain the knowledge of DIGITAL SYSTEM DESIGN AND ORGANIZATION 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 Multiplexer, Magnitude Comparators ,Code Converters etc. and implement by using logicgates/ICs.
- Present the working of various Flip-Flops, Register types, Counters.
- Understand the performance of computer and to execute the instructions.
- Analyze the basic organization and architecture of digital computers and number systems.

.

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-sumsMethod,Product-of-sumssimplifications,SimplificationbyQuine-McCluskyMethod.	09 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 Logic Unit, Clocks, Clock Waveforms	08 hours

Module -III Flip Flops: RS Flip-Flops, Gated Flip-Flops, Edge-triggered RS FLIP-FLOP. Clocked DFLIP-FLOP ,Edge-triggered D FLIP-FLOP, Edge- triggered JK FLIP-FLOP, FLIP-FLOP Timing,JK Master-slave FLIP-FLOP. Registers and Counters	08 hours
Module -IV 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. Input/OutputOrganization:AccessingI/ODevices,Interrupts–Interrupt Hardware, Enabling and Disabling Interrupts	08 hours
Module-V Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories–Mapping Functions, Replacement Algorithms, Performance Considerations. Arithmetic: Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers	09 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. Donald P Leach, Albert Paul Malvino & Goutam Saha: Digital Principles and Applications, 8thEdition, Tata McGraw Hill, 2015.

2. Carl Hamacher, Zvonko Vranesic, SafwatZaky: Computer Organization, 5th Edition,Tata McGraw Hill, 2011.

Reference Books:

1. Stephen Brown, Zvonko Vranesic: Fundamentals of Digital Logic Design with VHDL, 2ndEdition, Tata McGraw Hill,2012.
2.R.D.Sudhaker Samuel:IllustrativeApproachtoLogicDesign,SanguinePearson,2010.
3.WilliamStallings:ComputerOrganization&Architecture,7thEdition,PHI,2006

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, Registers and Counters
	CO4	Apply instruction sequencing to develop assembly language programs and implement stacks, queues and subroutines
	CO5	Analyze different storage devices, memory management and virtual memory concepts

Digital System Design and Organization LAB

Course Code	**22ISL35**	Credits:01
CIE:50	SEE:50	SEE:3hrs
Hours/Week:02(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 Digital System Design and Organization 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 using basic gates.
2. Design and Implement Half Subtractor, Full Subtractor using basic gates.
3. Simplify and realize the Boolean expression using logic gates.
4. Givenany4-variable logic expression, simplify using Entered Variable Map and realize the simplified logic using 8:1multiplexer IC.
5. Realize a full adder using3-to-8 decoder IC and 4 input NAND gates.
6. Design and implement code converter I) Binary to Gray II) Gray to Binary Code using basic gates.
7. Realize a J-K Master/Slave Flip-Flop using NAND gates and verify its truth table.
8. Design and implement a mod-n(n<8)synchronous up counter using J-K Flip-Flop Ics.
9. Design and implement a ring counter using 4-bit shift register.
10. Design and implement an asynchronous counter using decade counter IC to count up from 0 ton(n<=9).

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 Outcomes**
	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.

Discrete Mathematical Structure and Graph Theory		
Course Code	**22IS36A**	Credits:03
CIE:50	SEE:50	SEE:3hrs
Hours/Week:03		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 & Graph Theory in the following topics. • Understand and apply logic, relations, functions, basic set theory, count ability and counting arguments, proof techniques. •Understand and apply mathematical induction. •Understand various types of functions and operations on functions •Determine a connectivity of graph using the concepts of graph theory. •Understand and apply graph theory and mathematical proof techniques		
Modules		**Teaching Hours**
Module-I **Set Theory**: Sets and Subsets, Set Operations and the Laws of Set Theory, Countable and Uncountable Sets, The concept of probability , Mathematical Logic-I Propositions , Logical Equivalence and the law of logic, Rule of Inference .		9hours
Module-II **Mathematical Logic-II:** Open Statement Quantifiers , logical Implications involving Quantifiers, Methods of Proof and Disproof , Mathematical Induction. Relations and Functions: Cartesian Products and Relations, Functions – Types of Functions, Some Particular functions, Composition of Functions , Invertibe Functions, The Pigeon-hole Principle .		8 hours
Module –III **Relations II**: Zero-One Matrices and Directed Graphs, Operations on Relations, Properties of Relations , Equivalence Relations , Partial Orders-Total Order , External elements in Posets		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 coloring, chromatic Polynomials.	8 hours
Module-V **Trees:** Definitions , Properties and Examples, Rooted Trees, Tree and Sorting ,Weighted Trees and Prefix codes , Dijkstra's Shortest- Path Algorithm, Minimal Spanning Trees: The Algorithms of Kruskal and Prim	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. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education.2014.

2. Discrete Mathematical structures, Dr. D. S. Chandrashekariah. Prism

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. 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	Verify the correctness of an argument using propositional and predicate logic and truth tables
	CO2	Construct proofs using direct proof, proof by contraposition, proof by contradiction, proof by cases, and mathematical induction.
	CO3	Solve problems involving recurrence relations and generating functions

	CO4	Able to define the basic concepts of graphs, directed graphs, and weighted graphs to understand concept of coloring.
	CO5	Analyze various types of trees and tree traversing techniques

SOCIAL CONNECT AND RESPONSIBILITY

Course Code	**22UHV37**	Credits:01
CIE:50	SEE:00	SEE:--
Hours/Week:0-0-2		Total hours:28

Prerequisite: Nil

Course objectives:

- To familiarize the students with the basic biological concepts and their engineering applications.
- To enable the students with an understanding of bio design principles to create novel devices and structures.
- To provide the students an appreciation of how biological systems can be re-designed as substitute products for natural systems.
- To motivate the students to develop interdisciplinary vision of biological engineering.

Teaching-Learning Process(General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Explanation via real life problem, situation modeling, and deliberation of solutions, hands-on sessions, reflective and questioning /inquiry-based teaching.
2. Instructions with interactions in classroom lectures (physical/hybrid).
3. Use of ICT tools, including YouTube videos, related MOOCs, AR/VR/MR tools.
4. Flipped classroom sessions (~10% of the classes).
5. Industrial visits, Guests talks and competitions for learning beyond the syllabus.
6. Students' participation through audio-video based content creation for the syllabus (as assignments).
7. Use of gamification tools (in both physical/hybrid classes) for creative learning outcomes.
8. Students' seminars (in solo or group) /oral presentations

Modules	**Teaching Hours**
Module-I Plantation and adoption of a tree: Plantation of a tree that will be adopted for four years by a group of B.E students. They will also make an excerpt either as a documentary or a photo blog describing the plant's origin, its usage in daily life, and its appearance in folklore and literature.	6 hours

Objectives, Visit, case study, report, outcomes.	
Module-II Heritage walk and crafts corner: Heritage tour, knowing the history and culture of the city, connecting to people around through their history, knowing the city and its craftsman, photo blog and documentary on evolution and practice of various craft forms. Objectives, Visit, case study, report, outcomes.	06 hours
Module -III Organic farming and waste management: usefulness of organic farming, wet waste management in neighboring villages, and implementation in the campus. Objectives, Visit, case study, report, outcomes.	06 hours
Module -IV Water Conservation: knowing the present practices in the surrounding villages and implementation in the campus, documentary or photo blog presenting the current practices. Objectives, Visit, case study, report, outcomes.	05 hours
Module-V Food Walk City's culinary practices, food lore, and indigenous materials of the region used in cooking Objectives, Visit, case study, report, outcomes..	05 hours

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Communicate and connect to the surrounding. CO2: Create a responsible connection with the society
	CO2	Involve in the community in general in which they work.
	CO3	Notice the needs and problems of the community and involve them in problem –solving.

	CO4	Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.
	CO5	Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.

Activities: Jamming session, open mic, and poetry: Platform to connect to others. Share the stories with others. Share the experience of Social Connect. Exhibit the talent like playing instruments, singing, one-act play, art-painting, and fine art.

PEDAGOGY: The pedagogy will include interactive lectures, inspiring guest talks, field visits, social immersion, and a course project. Applying and synthesizing information from these sources to define the social problem to address and take up the solution as the course project, with your group. Social immersion with NGOs/social sections will be a key part of the course. Will all lead to the course project that will address the needs of the social sector?

COURSE TOPICS: The course will introduce social context and various players in the social space, and present approaches to discovering and understanding social needs. Social immersion and inspiring conversional will culminate in developing an actual, idea for problem-based intervention, based on an in-depth understanding of a key social problem.

Duration : A total of 40 - 50 hrs engagement per semester is required for the 3rd semester of the B.E. /B.Tech. program. The students will be divided into groups. Each group will be handled by faculty mentor. Faculty mentor will design the activities (particularly Jamming sessions open mic ,and poetry) Faculty mentors has to design the evaluation system as per VTU guidelines of scheme & syllabus.

Guideline for Assessment Process:
Continuous Internal Evaluation (CIE): After completion of the course, the student shall prepare, with daily diary as reference, a comprehensive report in consultation with the mentor/s to indicate what he has observed and learned in the social connect period. The report should be signed by the mentor. The report shall be evaluated on the basis of the following criteria and/or other relevant criteria pertaining to the activity completed. Marks allotted for the diary are out of 50. Planning and scheduling the social connect Information/Data collected during the social connect Analysis of the information/data and report writing Considering all above points allotting the marks as mentioned below
Excellent: 80 to 100
Good: 60 to 79
Satisfactory: 40 to 59
Unsatisfactory and fail : <39

Pedagogy-Guidelines
It may defer based on local resources available for study as well as environment and climatic differences, location, time of execution

Pedagogy-Guidelines

It may differ depending on local resources available for study as well as environment and climatic differences, location and time of execution.

s.no	Topic	Group Size	Location	Activity Execution	Reporting	Evaluation of the topic
1.	Plantation and adoption of a tree	May be individual or team	Farmers land, parks,villages,road side, community area/college campus etc	Site selection/proper consultation/continuous monitoring information board	Report should be submitted by the individual to the concerned evaluation authority	Evaluation as per the rubrics of the scheme and syllabus by Faculty
2	Heritage Walk and crafts corner	May be individual or team	Temples / monumental places / Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers/ campus etc…..	Site selection /proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
3	Organic farming and waste management:	May be individual or team	Farmers land / parks / Villages visits / roadside/ community area / College campus etc…..	Group selection / proper consultation / Continuous monitoring / Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
4	Water conservation: & conservation techniques	May be individual or team	Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers / campus etc….	site selection / proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluatio	Evaluation as per the rubrics Of scheme and syllabus by

					n authority	Faculty
5	Food walk: Practices in society	May be individual or team	Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers/ campus etc…..	Group selection / proper consultation / Continuous monitoring / Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty

Plan Of Action(Execution of Activities)

S.No	Practice Session Description
1	Lecture session in field to start activities
2	Students Presentation on Ideas
3	Commencement of activity and its progress
4	Execution of Activity
5	Execution of Activity
6	Execution of Activity
7	Execution of Activity
8	Case study based Assessment, Individual performance
9	Sector/ Team wise study and its consolidation
10	Video based seminar for 10 minutes by each student At the end of semester with Report

- Each student should do activities according to the scheme and syllabus.
- At the end of semester student performance has to be evaluated by the faculty for the assigned activity progress and its completion.
- At last consolidated report of all activities from 1st to 5th , compiled report should be submitted as per the instructions and scheme.

Assessment Details for CIE

Weight age	CIE – 100%	• Implementation strategies of the project (NSS work). • The last report should be signed by NSS Officer, the HOD and principal. • At last report should be evaluated by the NSS officer of the institute.
Field Visit, Plan, Discussion	10 Marks	
Commencement of activities and its progress	10	
Case study based Assessment Individual performance with report	10	
Sector wise study & its consolidation	10	
Video based seminar for 10 minutes by each student At the	10	

end of semester with Report. Activities 1 to 5		• Finally the consolidated marks sheet should be sent to the university and also to be made available at LIC visit
Total marks for the course in each semester	50 marks	
For each activity, 20 marks CIE will be evaluated for IA marks at the end of semester, Report and assessment copy should be made available in the department.		
Students should present the progress of the activities as per the schedule in the prescribed practical session in the field. There should be positive progress in the vertical order for the benefit of society in general through activities.		

DATA VISUALIZATION USING PYTHON

Course Code	**22ISAE381**	Credits:01
CIE:50	SEE:50	SEE:3hrs
Hours/Week:01		Total hours:15

Prerequisite: NIL

Course objectives: The Course will Enable students to

- Demonstrate the use of IDLE or PyCharm IDE to create Python Applications
- Using Python programming language to develop programs for solving real-world problems
- Implementation of Matplotlib for drawing different Plots
- Demonstrate working with Seaborn, Bokeh.
- Working with Plotly for 3D, Time Series and Maps.

List of Experiments

1.a) Write a python program to find the best of two test average marks out of three test's marks accepted from the user.

b) Develop a Python program to check whether a given number is palindrome or not andalso count the number of occurrences of each digit in the input number.

2.a) Defined as a function F as Fn = Fn-1 + Fn-2. Write a Python program which accepts a value for N (where N >0) as input and pass this value to the function. Display suitable error message if the condition for input value is not followed.

b) Develop a python program to convert binary to decimal, octal to hexadecimal using functions.

3.a) Write a Python program that accepts a sentence and find the number of words, digits, uppercase letters and lowercase letters.

b) Write a Python program to find the string similarity between two given strings

4.a) Write a Python program to Demonstrate how to Draw a Bar Plot using Matplotlib.

b) Write a Python program to Demonstrate how to Draw a Scatter Plot using Matplotlib.

5.a) Write a Python program to Demonstrate how to Draw a Histogram Plot using Matplotlib.
b) Write a Python program to Demonstrate how to Draw a Pie Chart using Matplotlib

6.a) Write a Python program to illustrate Linear Plotting using Matplotlib.
b) Write a Python program to illustrate liner plotting with line formatting using Matplotlib.

7. Write a Python program which explains uses of customizing seaborn plots with Aesthetic functions

8. Write a Python program to explain working with bokeh line graph using Annotations and Legends.
a) Write a Python program for plotting different types of plots using Bokeh.

9. Write a Python program to draw 3D Plots using Plotly Libraries

10. a) Write a Python program to draw Time Series using Plotly Libraries.
b) Write a Python program for creating Maps using Plotly Libraries.

Question paper pattern:

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

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Demonstrate the use of IDLE or PyCharm IDE to create Python Applications
	CO2	Use Python programming constructs to develop programs for solving real-world problems.
	CO3	Use Matplotlib for drawing different Plots

	CO4	Demonstrate working with Seaborn, Bokeh for visualization.
	CO5	Use Plotly for drawing Time Series and Maps.

NATIONAL SERVICE SCHEME

Course Code	**22NS39**	CIE:50
Semester:3	Credits NCMC – Non Credit Mandatory Course (Completion of the course shall be mandatory for the award of degree)	

SEE: Activities Report Evaluation by College NSS Officer at the end of every semester (3rd to 6th semester)

Course objectives:

National Service Scheme (NSS) will enable the students to:

1. Understand the community in general in which they work.
2. Identify the needs and problems of the community and involve them in problem –solving.
3. Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.
4. Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.
5. Develop capacity to meet emergencies and natural disasters & practice national integration and social harmony in general.

General Instructions - Pedagogy :

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the activities will develop students’ theoretical and applied social and cultural skills.
2. State the need for NSS activities and its present relevance in the society and Provide real-life examples.
3. Support and guide the students for self-planned activities.
4. You will also be responsible for assigning homework, grading assignments and quizzes, and documenting students’ progress in real activities in the field.
5. Encourage the students for group work to improve their creative and analytical skillsModules

Topics or activities to be covered

1 Organic farming, Indian Agriculture (Past, Present and Future) Connectivity for marketing.

2.Waste management– Public, Private and Govt organization, 5 R’s.

3.Setting of the information imparting club for women leading to contribution in social and economic issues

Suggested Learning material:

Books : 1. NSS Course Manual, Published by NSS Cell, VTU Belagavi.
2. Government of Karnataka, NSS cell, activities reports and its manual.
3. Government of India, nss cell, Activities reports and its manual.

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Understand the importance of his / her responsibilities towards society.
	CO2	Analyze the environmental and societal problems/issues and will be able to design solutions for the same
	CO3	Evaluate the existing system and to propose practical solutions for the same for sustainable development.
	CO4	Implement government or self-driven projects effectively in the field.
	CO5	Develop capacity to meet emergencies and natural disasters & practice national integration and social harmony in general.

PHYSICAL EDUCATION AND SPORTS		
Course Code 22PE39	CIE Marks 50	Credits 0
SEE Marks 00	Course Type Practical Lecture	Hours/Week (L-T-P) 0-0-3
Total Marks 50	Total Hours 28	Hours SEE Hours --

Guideline for Athletic and Sports			
Semester	Course Title	Content	No. of Hours
3rd sem	Fitness Components Speed Strength Endurance Agility Flexibility	Meaning and Importance, Fit India Movement, Definition of fitness, Components of fitness, Benefits of fitness, Types of fitness and Fitness tips. Practical Components: Speed, Strength, Endurance, Flexibility, and Agility KABADDI A. Fundamental skills 1. Skills in Raiding: Touching with hands, Use of leg-toe touch, squat leg thrust, side kick, mule kick, arrow fly kick, crossing of baulk line. Crossing of Bonus line. 2. Skills of holding the raider: Various formations, catching from particular position, different catches, catching formation and techniques. 3. Additional skills in raiding: Escaping from various holds, techniques of escaping from chain formation, offense and defense. 4. Game practice with application of Rules and Regulations. B. Rules and their interpretations and duties of the officials.	Total 28 hrs 2 hrs / week
	Kho Kho	A. Fundamental skills 1. Skills in Chasing: Sit on the box (Parallel & Bullet toe method), Get up from the box (Proximal & Distal foot method), Give Kho (Simple, Early, Late & Judgment), Pole Turn, Pole Dive, Tapping, Hammering, Rectification of foul. 2. Skills in running: Chain Play, Ring play and Chain & Ring mixed play. 3. Game practice with application of Rules and Regulations. B. Rules and their interpretations and duties of the officials.	

	Kabaddi	A. Fundamental skills 1. Skills in Raiding: Touching with hands, Use of leg-toe touch, squat leg thrust, side kick, mule kick, arrow fly kick, crossing of baulk line. Crossing of Bonus line. 2. Skills of holding the raider: Various formations, catching from particular position, different catches, catching formation and techniques. 3. Additional skills in raiding: Escaping from various holds, techniques of escaping from chain formation, offense and defense. 4. Game practice with application of Rules and Regulations. B. Rules and their interpretations and duties of the officials	

REFERENCES

1. Saha, A. K. Sarir Siksher Ritiniti, Rana Publishing House, Kalyani.
2. Bandopadhyay, K. Sarir Siksha Parichay, Classic Publishers, Kolkata.
3. Petipus, et al. Athlete’s Guide to Career Planning, Human Kinetics.
4. Dharma, P.N. Fundamentals of Track and Field, Khel Sahitya Kendra, New Delhi.
5. Jain, R. Play and Learn Cricket, Khel Sahitya Kendra, New Delhi.
6. Vivek Thani, Coaching Cricket, Khel Sahitya Kendra, New Delhi.
7. Saha, A. K. Sarir Siksher Ritiniti, Rana Publishing House, Kalyani.
8. Bandopadhyay, K. Sarir Siksha Parichay, Classic Publishers, Kolkata
9. Naveen Jain, Play and Learn Basketball, Khel Sahitya Kendra, New Delhi.
10. Dubey, H. C. Basketball, Discovery Publishing House, New Delhi.
11. Rachana Jain, Teach Yourself Basketball, Sports Publication.
12. Jack Nagle, Power Pattern Offences for Winning asketball, Parker Publishing Co., New York.
13. Renu Jain, Play and Learn Basketball, Khel Sahitya Kendra, New Delhi.
14. Sally Kus, Coaching Volleyball Successfully, Human Kinetics.
15. Saha, A. K. Sarir Siksher Ritiniti, Rana Publishing House, Kalyani. 16 Bandopadhyay, K. Sarir
Siksha Parichay, Classic Publishers, Kolkata
16. Test and Measurement (by Cleark and Cleark)
17. Evaluation in Physical Education (by Dr. Devendraya Kausal)
18. Methods of Physical Education (by Haridrash & Prof. Tirumalay Swamy)
19. Athletics (by Hardayal Singh)
20. Efficienting and Coaching (by Dr. Anand Nadigri)
21. Modern and Ancient History of Physical Education (by Dr. D. M. Jyothi)
22. Organization and Administration (by K. G. Nadigir or Vastrad

YOGA AND MEDITATION

Course Code	**22YO39**	Credits:0
CIE:50	SEE: 00	SEE:--
Hours/Week: (L:T:P: S):0-0-3(Practical)	Total Marks:50	Total hours:28 hours

Prerequisite:NIL

Course objectives: The Course will Enable students to

- To enable the student to have good health.
- To practice mental hygiene
- To possess emotional stability.
- To integrate moral values.
- To attain higher level of consciousness.

Semester	Content
III	1) Introduction of Yoga, Aim and Objectives of yoga, Prayer 2) Brief introduction of yogic practices for common man 3) Rules and regulations 4) Misconceptions of yoga 5) Suryanamaskara 6) Different types of Asanas a. Sitting b. Standing c. Prone line d. Supine line

Guidelines

Semester	**Course Title**	**Content**	No.of Hours
3rd Semester	Introduction to Yoga,Aim and Objectives of Yoga Prayer	Yoga,its origin,history and development,yoga its meaning definition,different	2hrs/week

		schools of yoga,importance of prayer	
	Brief Introduction of yogic practices for common man	yogic practices for common man to promote positive health	
	Rules and Regulations	Rules to be followed during yogic practices by practioner	
	Misconceptions of yoga	Yoga its Misconceptions,Difference between yogic and non yogic practices	
	SuryaNamaskar	SuryaNamaskar,Prayer and its meaning,Need,importance and benefits of SuryaNamaskar count 12, 2 rounds	
	Different Types of Asanas Sitting i)Padmasana ii)Vajrasana Standing i)Vrikshasana ii)Trikonasana prone Line i)Bhujangasana ii)Shalabhasana Supine Line i)Utthita Dwipadasana	Asanas, Need Importance of Asanas,Different types of Asana,Asana its meaning by name, technique, precautionary measure and benefits of each asana	

	ii)Ardha Halasana		

Suggested Books

·Swami Kuvulyananda : Asma (Kavalyadhama, Lonavala)
·Tiwari, O P : Asana Why and How
·Ajitkumar : Yoga Pravesha (Kannada)
·Swami Satyananda Saraswati: Asana Pranayama, Mudra, Bandha (Bihar School of yoga, Munger)
·Swami Satyananda Saraswati : Surya Namaskar (Bihar School of yoga, Munger)
·Nagendra H R : The art and science of Pranayama
·Tiruka : Shatkriyegalu (Kannada)
·Iyengar B K S : Yoga Pradipika (Kannada)
·Iyengar B K S : Light on Yoga (English)

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Understand the meaning, aim and objectives of Yoga.
	CO2	Perform Suryanamaskar and able to teach its benefits.
	CO3	Understand and teach different Asanas by name, its importance, methods and benefits
	CO4	Instruct Kapalabhati and its need and importance
	CO5	Teach different types of Pranayama by its name, precautions, procedure and uses
	CO6	Coach different types of Kriyas , method to follow and usefulness

P D A College of Engineering

B.E. Information Science & Engineering

Scheme of Teaching and Examination 2022

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

(Effective from the academic year2023-24)

IV SEMESTER

Sl. No	**Course and Course Code**		**Course Title**	**TeachingDepartment (TD)and QuestionPaper SettingBoard(PSB)**	Teaching Hours/Week				Examination				**Credits**
					TheoryL ecture	TD: **Tutorial**	**Practical/D**	TD: **Self-Study**	**Duration inhours**	**CIEMarks**	**SEEMarks**	**TotalMarks**	
					L	**T**	**P**	**S**					
1	PCC	22IS41	Analysis and Design of Algorithms	TD:ME PSB:ME	2	2	0		03	50	50	100	3
2	IPCC	22IS42	Application Development Using Java	TD-RespectiveDept. PSB- RespectiveDept.	3	0	2		03	50	50	100	4
3	IPCC	22IS43	Microcontrollers & Embedded Systems	TD-RespectiveDept. PSB-RespectiveDept.	3	0	2		03	50	50	100	4
4	PCCL	22ISL44	Analysis and Design of Algorithms Lab	TD-RespectiveDept. PSB-RespectiveDept.	0	0	2		03	50	50	100	1
5	ESC	22IS45A	Automata Theory and Computability	Respective Dept. PSB:RespectiveDept.	3	0	0		03	50	50	100	3
6	BSC	22BSC46	Biology For Engineers	TD/PSB:BT, CHE,	3	0	0		03	50	50	100	3
7	UHV	22UHV47	Universal Human Values	Any Department	1	0	0		02	50	50	100	1
8	AEC/S EC	22ISAE481	Ability Enhancement Course/Skill Enhancement Course-IV Internet of Things	TD and PSB:Concerneddepart ment	If the course is Theory				02	50	50	100	1
					0	0	2						
					If the course is lab				03				
					0	0	2						
9	NCMC	22NS49	Mandatory Course	NSScoordinator	0	0	2			50	- - -	50	0
		22PE49	Mandatory Course	PhysicalEducation Director									
		22YO49	Mandatory Course	YogaTeacher									
									Total	**450**	**400**	**850**	**20**

PCC:ProfessionalCoreCourse,**PCCL**:ProfessionalCoreCourselaboratory,**UHV**:UniversalHumanValueCourse,**MC**:MandatoryCourse(Non-credit),**AEC**:AbilityEnhancement Course, **SEC**:Skill Enhancement Course,**L:**Lecture,**T**:Tutorial,**P**:Practical**S=SDA**:Skill Development Activity,**CIE**:Continuous Internal Evaluation,**SEE**:
Semester End Evaluation. K: This letter in the course code indicates common to all the stream of engineering.
Engineering Science Course: The course is not common to all the departments and it is relevant to the respective departments

Professional Core Course (IPCC): Refers to Professional Core Course Theory Integrated with practical of the same course. Credit for IPCC can be 04 and its Teaching–Learning hours (L : T : P) can be considered as(3 : 0 : 2) or (2 : 2 : 2). The theory part of the IPCC shall be evaluated both by CIE and SEE. The practical part shall beevaluatedbyonlyCIE(noSEE).However,questionsfromthepracticalpartofIPCCshallbeincludedintheSEEquestionpaper.Formoredetails,theregulationgoverningtheDegreeofBachelorofEngineering/Technology(B.E./B.Tech.)2022-23

National Service Scheme /Physical Education/Yoga: All students have to register for any one of the courses namely National Service Scheme (NSS), Physical Education (PE) (Sports and Athletics), and Yoga(YOG) with the concerned coordinator of the course during the first week of III semesters. Activities shall be carried out between III semester to the VI semester (for 4 semesters). Successful completion of the registered course and requisite CIE score is mandatory for the award of the degree. Theeventsshallbeappropriatelyscheduledbythecollegesandthesameshallbereflectedin the calendar prepared for the NSS, PE and Yoga activities. These courses shall not be considered for vertical progression as well as for the calculation of SGPA and CGPA, but completion of the courses is mandatory for the award of degree.

Ability Enhancement Course /Skill Enhancement Course(AEC/SEC)			
22ISAE481	Internet Of Things	22ISAE482	MERN
22ISAE483	Scala	22ISAE484	Julia

HKE Society’s

PDA COLLEGE OF ENGINEERING, KALABURAGI

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

B.E. IV SEMESTER

ANALYSIS AND DESIGN OF ALGORITHMS

Course Code	**22IS41**	Credits:03
CIE:50	SEE:50	SEE:3hrs
Hours/Week:03		Total hours:42

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

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	8 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	9 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 TENquestions.
2.There will be TWO questions in each module, covering all thetopics.
3.The student need to answer FIVE full questions, selecting ONE full question from each module.

Text books:

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.

APPLICATION DEVELOPMENT USING JAVA		
Course Code	**22IS42**	Credits:04
CIE:50	SEE:50	SEE:3hrs
Hours/Week:04(T+L)		Total hours:52(40+12)
Prerequisite: The students should have the thorough knowledge of Object Oriented andProcedure Oriented Programming Paradigm		
Course Objectives: To enable the students to obtain the knowledge of JAVA in the following topics. • Understand the concepts of exception handling and Event Handling Mechanism. • Understand the importance of Packages and Multithreading Concepts. • Understand Stream Handling Mechanism and Handling I/O Files. • Understand the Programming Principles of Applet programming and Implementing Applications using JAVA Principals.		
Modules		**Teaching Hours**
Module-I **Introduction to JAVA:** Overview of JAVA, Java applications, JDK, Compiling Java Program, Java Interpreter, Byte code, JVM, Simple JAVA Programs. Primitive, non-primitive data types, Type casting, Arrays and strings. **Operators & Expressions:** Arithmetic operators, Bitwise operators, Relational Operators, Logical Operators, The Assignment Operators, The? : Operators, Operator precedence; Logical expression; Control statements, Selection statements, Iteration statements, Jump statements.		8 hours
Module-II Class, Objects, Methods: Classes in Java, Class fundamentals, Super classes, Constructors; Creating instances of class; Methods; Method overloading. **Inheritance** : Simple, Multiple and multilevel inheritance, overriding, overloading, using abstract classes, using final with inheritance.		8 hours
Module -III **Packages:** Creating package, Access package, importing package; defining Interfaces, implanting interfaces, Accessing interface variables. **Exception Handling:** Exception type, Multiple catch statements, uncaught exceptions, using try and catch block, Nested try statements, Multiple catch statements Java built in exceptions.		8 hours

Module -IV **Event Handling:** Event handling mechanisms, The delegation event model, event classes, source of events, Event listener interfaces, Adapter classes, inner classes. **Multithread Programming :** Java thread model, thread priorities, Synchronization, Messaging, thread class and run able interface, main thread, creating a thread, multiple, threads, stopping and blocking a thread, Thread life cycle, thread methods, thread exceptions .	8 hours
Module-V **Applet Programming :** The Applet Class: Applet basics, Two types of Applets; Applet Architecture; An Applet skeleton; Applet lifecycle, Simple Applet display methods; Requesting repainting; Using the Status Window; Designing the web page, The HTML APPLET tag; Adding applet to HTML File, Passing parameters to the APPLETS; getDocumentbase() and showDocument( ), The AudioClip Interface; The AppletStub Interface; Output to the Console. **Managing I/O Files in JAVA:** Stream classes, byte stream classes, character stream classes, other I/O classes, I/O exceptions, Reading writing character, Reading writing bytes. Other stream classes.	8 hours

List of Experiments

1. Write a program using do-while loop to calculate and print the first m Fibonacci numbers. (Hint: After the first two numbers in the series, each number is the sum of the two preceding numbers).

2.Write a program to print the following outputs using for loops

```
      1               1
      2 2            2 2
    3 3 3          3 3 3
   4 4 4 4       4 4 4  4
                             5    5 5 5 5 5 5 5  5 5
```

3.Write a program which will read at extend count all occurrences of a particular word

4. Write a Java program to create class ACC with data members,accno, balance. Create objects ACC1, ACC2 & ACC3. Write a member functiontotransfermoneyfromACC3toACC1,display the balance in all accounts

5. Write a Java program to implement the concept of multiple inheritance using interfaces.

6. Write a program to create an interface variable and access stacks through it.

7. Write a Java program for handling mouse events.

8.Develop an applet that receives an integer in one text field, and computes its factorial value and returns it in another text field, when button name “compute” is clicked.

9.Write an applet program for menu demonstration, menu bar should contain File, Edit, View and its sub menus.

10.Write an applet program for key event sit should recognize normal as well as special keys and should be displayed on the panel.

11. Write a Java program that creates three threads .First thread displays “ Good Morning” every one second, the second thread displays “ hello’ every two seconds and the third thread displays “Welcome” every three seconds
12. Write a java program that illustrate the suspend, resume and stop operations in thread.
13. Write a java program that illustrates nested try statements.
14. Write a java program to illustrate the use of access control modifiers on two packages.
15. Write a java program to store and retrieve integers using data streams on a single file.

PART B

Case Studies:

1.Grading System in JAVA

2. School management System

3.CGPA Calculation in JAVA

4. Simple calculator using JAVA

5.Address book using JAVA

6.Temperature Converter using JAVA

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. Java the Complete Reference - Herbert Schildt, 7th Edition, Tata McGraw Hill, 2007.
2. Programming with Java 5th Edition – E. Balaguruswamy, Tata McGraw Hill.

Reference Books:

1. Introduction to JAVA Programming - Y. Daniel Liang, 12th Edition, Pearson Education.

2. Introduction to JAVA Programming - Y. Daniel Liang, 6th Edition, Pearson Education, 2007.

E-Books and Course Materials

Thinking in JAVA Author : Bruce Eckel
Download Link: http://www.mindview.net/Books/TIJ

The JAVA Language Specification, Author: James Gosling, Bill Joy, Guy Steele, Gilad Bracha, and Alex Buckley.

Read Online: http://docs.oracle.com/javase/specs/jls/se8/html/index.html
Download PDF: http://docs.oracle.com/javase/specs/jls/se8/jls8.pdf **Publish Date:** March 2014

The JAVA Tutorials
Author: Raymond Gallardo, Scott Hommel, Sowmya Kannan, Joni Gordon, and Sharon BioccaZakhour.
Read Online: http://docs.oracle.com/javase/tutorial
Download Link: http://www.oracle.com/technetwork/java/javase/java-tutorial-downloads-2005894.html, **Publish Date:** August 2014

Think JAVA
Author: Allen B. Downey
Read Online: http://greenteapress.com/thinkapjava/html/index.html
Download PDF: http://greenteapress.com/thinkapjava/thinkapjava.pdf
Publish Date: July 2011 (5th edition).

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Analyze and implement the OOP principles using class and objects.
	CO2	Implement the inheritance modules using JAVA principles
	CO3	Analyzing the built-in packages, exceptions and event handlingmechanism
	CO4	Applying the multithreading and applet programming principles todesign JAVA based applications.
	CO5	Analyzing the stream handling mechanism and implementing the real time JAVA applications.

MICROCONTROLLER AND EMBEDDED SYSTEMS

Course Code	**22IS43**	Credits:04
CIE:50	SEE:50	SEE:3hrs
Hours/Week:04(T+L)		Total hours:52 (40+12)

Prerequisite: Students Should have knowledge of logic gates

Course Learning Objectives:

To enable the students to obtain the knowledge of Microcontroller and Embedded Systems in the following topics.

- Understand the fundamentals of ARM-based systems, including programming modules with registers and the CPSR.
- Use the various instructions to program the ARM controller.
- Identify various components, their purpose, and their application to the embedded system's applicability.
- Understand the embedded system's real-time operating system and its application

Modules	**Teaching Hours**
Module-I 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, Core Extensions Textbook 1: Chapter 1 - 1.1 to 1.4, Chapter 2 - 2.1 to 2.5	08 hours
Module-II Introduction to the ARM Instruction Set: Data Processing Instructions, Branch Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions, Loading Constants C Compilers and Optimization: Basic C Data Types, C Looping Structures, Register Allocation, Function Calls, Pointer Aliasing. Textbook 1: Chapter 3: Sections 3.1 to 3.6 (Excluding 3.5.2), Chapter 5	08 hours
Module -III C Compilers and Optimization: Structure Arrangement, Bit-fields, Unaligned Data and Endianness, Division, Floating Point, Inline Functions and Inline Assembly, Portability Issues. ARM programming using Assembly language: Writing Assembly code, Profiling and cycle counting, instruction scheduling, Register Allocation, Conditional Execution, Looping Constructs.	08 hours

Textbook 1: Chapter-5,6	
Module -IV Embedded System Components: Embedded Vs General computing system, History of embedded systems, Classification of Embedded systems, Major applications areas of embedded systems, purpose of embedded systems. Core of an Embedded System including all types of processor/controllers, Memory, Sensors, Actuators, LED, 7 segment LED display, stepper motor, Keyboard, Push button switch, Communication Interface (onboard and external types), Embedded firmware, other system components. Textbook 2: Chapter 1 (Sections 1.2 to 1.6), Chapter 2 (Sections 2.1 to 2.6)	08 hours
Module-V RTOS and IDE for Embedded System Design: Operating System basics, Types of operating systems, Task, process and threads (Only POSIX Threads with an example program), Thread preemption, Multiprocessing and Multitasking, Task Communication (without any program), Task synchronization issues – Racing and Deadlock, Concept of Binary and counting semaphores (Mutex example without any program), How to choose an RTOS, Integration and testing of Embedded hardware and firmware, Embedded system Development Environment – Block diagram (excluding Keil), Disassembler/decompiler, simulator, emulator and debugging techniques, target hardware debugging, boundary scan. Textbook 2: Chapter-10 (Sections 10.1, 10.2, 10.3, 10.4 , 10.7, 10.8.1.1, 10.8.1.2, 10.8.2.2, 10.10 only), Chapter 12, Chapter-13 (block diagram before 13.1, 13.3, 13.4, 13.5, 13.6 only)	08 hours

List of Experiments

1. Using Keil software, observe the various registers, CPSR, with a simple ALP program.
2. Write a program to find the sum of the first 10 integer numbers.
3. Write a program to find the factorial of a number.
4. Write a program to add an array of 16 bit numbers and store the 32 bit result in internal RAM.
5. Write a program to find the square of a number (1 to 10) using a look-up table.
6. Write a program to find the largest or smallest number in an array of 32 numbers.
7. Write a program to arrange a series of 32 bit numbers in ascending/descending order.
8. Write a program to count the number of ones and zeros in two consecutive memory locations.
9. Display "Hello World" message using Internal UART.
10. Interface and Control a DC Motor.
11. Interface a Stepper motor and rotate it in clockwise and anti-clockwise direction.
12. Interface a DAC and generate Triangular and Square waveforms.
13. Interface a 4x4 keyboard and display the key code on an LCD.
14. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.

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.Andrew NS loss, Dominic Symes and Chris Wright, ARM system developers guide, Elsevier, Morgan Kaufman publishers, 2008.

2.Shibu K V, "Introduction to Embedded Systems", Tata McGraw Hill Education, Private Limited,2ndEdition.

Reference Books:

1. Raghu Nandan ..G.H,Microcontroller (ARM) and Embedded System, Cengage learning Publication, 2019
2. The Insider's Guide to the ARM7 Based Microcontrollers,Hitex Ltd,Ist Edition 2005 .
3. SteveFurber,ARMSystem-on-ChipArchitecture,SecondEdition,Pearson,2015.
4. RajKamal,EmbeddedSystem,TataMcGraw-HillPublishers,2ndEdition,2008

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Explain C-Compilers and optimization
	CO2	Describe the ARM microcontroller's architectural features and program module.
	CO3	Apply the knowledge gained from programming on ARM to different applications.
	CO4	Program the basic hardware components and their application selection method.
	CO5	Demonstrate the need for a real-time operating system for embedded system applications.

ANALYSIS AND DESIGN OF ALGORITHMS LAB

Course Code	**22ISL44**	Credits:01
CIE:50	SEE:50	SEE:3hrs
Hours/Week:02(Practical)		Total hours:28

Prerequisite: The students must have the knowledge of C, Data Structures 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

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.
10.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 Outcomes**
	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.

<table>
<tr><th colspan="3">AUTOMATA THEORY AND COMPUTABILITY</th></tr>
<tr><td>Course Code</td><td><b>22IS45A</b></td><td>Credits:03</td></tr>
<tr><td>CIE:50</td><td>SEE:50</td><td>SEE:3hrs</td></tr>
<tr><td colspan="2">Hours/Week:03</td><td>Total hours:42</td></tr>
<tr><td colspan="3">Prerequisite: : The students should have good knowledge of discrete mathematical structures, data structures, programming principles and computer architecture</td></tr>
<tr><td colspan="3">Course objectives: To enable the students to obtain the knowledge of Automata Theory and Computability in the following topics
•Introduce core concepts in Automata and Theory of Computation to design automata generating a certain language.
•Design regular expression and identify different form all Language Classes and their relationships
•Design grammars and recognizers for different form all languages and translate between deterministic and non deterministic pushdown automata.
•Define Turing machines performing simple tasks to prove or disprove theorems in automata theory using their properties
•Determine the decidability and intractability of Computational problems</td></tr>
<tr><th colspan="2">Modules</th><th>Teaching Hours</th></tr>
<tr><td colspan="2"><b>Module-I</b>
<b>Introduction-</b> Introduction, Basic Notations and Terminologies used ,Finite Automata, DFA, DFA Design Techniques, Applications of DFA, Non-Deterministic Finite Automata, Conversion from NFA to DFA.</td><td>9 hours</td></tr>
<tr><td colspan="2"><b>Module-II</b>
<b>Finite Automata and Regular Expression (RE)- Finite automata with Epsilon Transitions , Conversion from Є- NFA to DFA,</b> Regular expression (RE) Definition, Finite automata and Regular Expressions, Applications of Regular expressions, Proving languages not to be regular, Properties of Regular Languages, Equivalence and Minimization of Finite Automata.</td><td>9 hours</td></tr>
<tr><td colspan="2"><b>Module -III</b>
<b>Context Free Grammar (CFG) and Context Free Languages (CFL)-</b> Definition, Examples, Derivation, Derivation trees, Ambiguous Grammar, left Recursion, Simplification of CFG, Eliminating Є- productions, Eliminating Unit Productions, Chomsky Normal Form, Greibach Normal Form, Pumping Lemma, CFLs are closed under Union , Concatenation and star, CFLs are not closed under intersection and complementation.</td><td>8 hours</td></tr>
</table>

Module -IV **Push Down Automata (PDA)-** Transitions , Graphical Representation of PDA, Instantaneous Description, Acceptance of a language by PDA, Construction of PDA, Deterministic and Non- deterministic PDA , CFG to PDA , PDA to CFG.	8 hours
Module-V **Turing machines (TM)-** Turing Machine Model, Representation of Turing Machines, Design Of Turing Machines, Techniques of TM construction, Variants of Turing Machines, **Decidability, Decidable languages, Undecidable languages, Halting problem of TM, The post Correspondence problem, The classes of P and NP, Quantum Computation.**	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. K L P Mishra, N Chandrasekaran , 3rd Edition, Theory of Computer Science, PhI, 2012.

2. Finite Automata and Formal Languages A simple Approach- A . M .Padma Reddy

3. Hopcroft and Ullman, “Introduction to Automata Theory, Languages and Computation”, Pearson Education, 3rd edition, 2006

Reference Books:

1. John E Hopcroft, Rajeev Motwani, Jeffery D Ullman, Introduction to Automata Theory, Languages, and Computation,3rd Edition, Pearson Education, 2013
2. Michael Sipser:Introduction to theTheory of Computation,3rdedition,Cengage learning,2013
3. JohnC Martin,Introduction toLanguagesand TheTheoryofComputation, 3rdEdition, TataMcGraw–HillPublishingCompanyLimited,2013
4. PeterLinz,“An Introduction to Formal Languages and Automata”,3rd Edition, Narosa Publishers, 1998
5. Basavaraj S.Anami,KaribasappaKG,FormalLanguagesandAutomatatheory,WileyIndia,2012

CK Nagpal, Formal Languages and AutomataTheory,OxfordUniversitypress,2012.

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Attain fundamental understanding of the core concepts in automata theory and theory of computation

	CO2	Illustrate how to translate between different models of Computation
	CO3	Design grammars and automata (recognizers) for different language classes and become familiar about restricted models of Computation and their relative powers
	CO4	Develop skills in formal reasoning and reduction of a problem to a formal model, with an emphasis on semantic precision and conciseness
	CO5	Categorize a problem with respect to different models of Computation.

BIOLOGY FOR ENGINEERS

Course Code	**22BSC46**	Credits:03
CIE:50	SEE:50	SEE:3hrs
Hours/Week:03		Total hours:42

Prerequisite: : NIL

Course objectives: To familiarize the students with the basic biological concepts and their engineering Applications.

- To enable the students with an understanding of bio design principles to create novel devices and structures.
- To provide the students an appreciation of how biological systems can be re- designed as substitute products for natural systems.
- To motivate the students to develop interdisciplinary vision of biological engineering.

Teaching-Learning Process (General Instructions)

These are sample Strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Explanation via real life problem, situation modelling, and deliberation of solutions, hands-on sessions, reflective and questioning /inquiry-based teaching.
2. Instructions with interactions in classroom lectures (physical/hybrid).
3. Use of ICT tools, including YouTube videos, related MOOCs, AR/VR/MR tools.
4. Flipped classroom sessions (~10% of the classes).
5. Industrial visits, Guests talks and competitions for learning beyond the syllabus.
6. Students' participation through audio-video based content creation for the syllabus (as assignments).
7. Use of gamification tools (in both physical/hybrid classes) for creative learning outcomes. 8. Students' seminars (in solo or group) /oral presentations

Modules	**Teaching Hours**
Module-I Introduction To Biology: The cell: the basic unit of life, Structure and functions of a cell. The Plant Cell and animal cell, Prokaryotic and Eukaryotic cell, Stem cells and their application. Biomolecules: Properties and functions of Carbohydrates, Nucleic acids, proteins, lipids. Importance of special biomolecules; Enzymes (Classification (with one example each),Properties and functions), vitamins and hormones.	08 hours

Module-II Biomolecules And Their Applications (Qualitative) : Carbohydrates (cellulose-based water filters, PHA and PLA as bioplastics), Nucleic acids (DNA Vaccine for Rabies and RNA vaccines for Covid19, Forensics – DNA fingerprinting), Proteins (Proteins as food – whey protein and meat analogs, Plant based proteins), lipids (biodiesel, cleaning agents/detergents), Enzymes (glucoseoxidase in biosensors, lignolytic enzyme in bio-bleaching).	09 hours
Module -III Human Organ Systems And Bio Designs (Qualitative): Brain as a CPU system (architecture, CNS and Peripheral Nervous System, signal transmission, EEG, Robotic arms for prosthetics. Engineering solutions for Parkinson's disease).Eye as a Camera system (architecture of rod and cone cells, optical corrections, cataract, lens materials, bionic eye). Heart as a pump system (architecture, electrical signalling - ECG monitoring and heart related issues, reasons for blockages of blood vessels, design of stents, pace makers, defibrillators). Lungs as purification system (architecture, gas exchange mechanisms, spirometry, abnormal lung physiology - COPD, Ventilators, Heart-lung machine). Kidney as a filtration system (architecture, mechanism of filtration, CKD, dialysis systems).	08 hours
Module -IV Nature-Bioinspired Materials And Mechanisms (Qualitative): Echolocation (ultrasonography, sonars), Photosynthesis (photovoltaic cells, bionic leaf). Bird flying (GPS and aircrafts), Lotus leaf effect (Super hydrophobic and self-cleaning surfaces), Plant burrs (Velcro), Shark skin (Friction reducing swim suits), Kingfisher beak (Bullet train). Human Blood substitutes - hemoglobin-based oxygen carriers (HBOCs) and perflourocarbons (PFCs)	09 hours
Module-V Trends In Bioengineering (Qualitative): Muscular and Skeletal Systems as scaffolds (architecture, mechanisms, bioengineering solutions for muscular dystrophy and osteoporosis), scaffolds and tissue engineering, Bioprinting techniques and materials, 3D printing of ear, bone and skin. 3D printed foods. Electrical tongue and electrical nose in food science, DNA origami and Biocomputing, Bioimaging and Artificial Intelligence for disease diagnosis. Self healing Bioconcrete (based on bacillus spores, calcium lactate nutrients and biomineralization processes) and Bioremediation and Biomining via microbial surface adsorption (removal of heavy metals like Lead, Cadmium, Mercury, Arsenic).	08 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.

Suggested Learning Resources

Text books:

1. Biology for Engineers, Rajendra Singh C and Rathnakar Rao N, Publishing, Bengaluru, 2023.·

2. Human Physiology, Stuart Fox, Krista Rompolski, McGraw-Hill eBook. 16th Edition, 2022·

3. Biology for Engineers, Thyagarajan S., Selvamurugan N., Rajesh M.P., Nazeer R.A., Thilagaraj W., Barathi S., and Jaganthan M.K., Tata McGraw-Hill, New Delhi, 2012.·

4. Biology for Engineers, Arthur T. Johnson, CRC Press, Taylor and Francis, 2011·

5. Biomedical Instrumentation, Leslie Cromwell, Prentice Hall 2011.·

6. Biology for Engineers, Sohini Singh and Tanu Allen, Vayu Education of India, New Delhi, 2014.·

7. Biomimetics: Nature-Based Innovation, Yoseph Bar-Cohen, 1st edition, 2012, CRC Press.

8. Bio-Inspired Artificial Intelligence: Theories, Methods and Technologies, D. Floreano and C. Mattiussi, MIT Press, 2008.·

9. Bioremediation of heavy metals: bacterial participation, by C R Sunilkumar, N Geetha A C Udayashankar Lambert Academic Publishing, 2019.·

10. 3D Bioprinting: Fundamentals, Principles and Applications by Ibrahim Ozbolat, Academic Press, 2016.

11. Electronic Noses and Tongues in Food Science, Maria Rodriguez Mende, Academic Press, 2016

Web links and Video Lectures (e-Resources):

- https://nptel.ac.in/courses/121106008
- https://freevideolectures.com/course/4877/nptel-biology-engineers-other-non-biologists
- https://ocw.mit.edu/courses/20-020-introduction-to-biological-engineering-design-spring2009
- https://ocw.mit.edu/courses/20-010j-introduction-to-bioengineering-be-010j-spring-2006
- https://www.coursera.org/courses?query=biology
- https://onlinecourses.nptel.ac.in/noc19_ge31/preview
- https://www.classcentral.com/subject/biology
- https://www.futurelearn.com/courses/biology-basic-concepts.

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

Group Discussion of Case studies

- Model Making and seminar/poster presentations
- Design of novel device/equipment like Cellulose-based water filters, Filtration system

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**

	CO1	Elucidate the basic biological concepts via relevant industrial applications and case studies.
	CO2	Evaluate the principles of design and development, for exploring novel bioengineering projects
	CO3	Corroborate the concepts of biomimetics for specific requirements
	CO4	Think critically towards exploring innovative biobased solutions for socially relevant problems

UNIVERSAL HUMAN VALUES		
Course Code	**22UHV47**	Credits:01
CIE:50	SEE:50	SEE:2 hrs
Hours/Week:02		Total hours:15
Prerequisite: : The students should have good knowledge of discrete mathematical structures, data structures, programming principles and computer architecture		
Course objectives: • To help the students appreciate the essential complementarity between 'VALUES' and' SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings. • To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence. Such a holistic perspective forms the basis of Universal Human Values and movement towards value-based living in a natural way. • To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature. • This course is intended to provide a much-needed orientation input in value education to the young enquiring minds.		
Modules		**Teaching Hours**
Introduction to Value Education		3 hours

Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education) Understanding Value Education, Self-exploration as the Process for Value Education, Continuous Happiness and Prosperity – the Basic Human Aspirations, Happiness and Prosperity – Current Scenario, Method to Fulfill the Basic Human Aspirations	
Module-II **Harmony in the Human Being** Understanding Human being as the Co-existence of the Self and the Body, Distinguishing between the Needs of the Self and the Body, The Body as an Instrument of the Self Understanding Harmony in the Self, Harmony of the Self with the Body, Program to ensure self-regulation and Health	3 hours
Module -III **Harmony in the Family and Society** : (3 hours) Harmony in the Family – the Basic Unit of Human Interaction, 'Trust' – the Foundational Value in Relationship, 'Respect' – as the Right Evaluation, Other Feelings, Justice in Human-to Human Relationship, Understanding Harmony in the Society, Vision for the Universal Human Order.	3 hours
Module -IV **Harmony in the Nature/Existence** : (3 hours) Understanding Harmony in the Nature, Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature, Realizing Existence as Co-existence at All Levels, The Holistic Perception of Harmony in Existence	3 hours
Module-V **Implications of the Holistic Understanding** – a Look at Professional Ethics Natural Acceptance of Human Values, Definitiveness of (Ethical) Human Conduct, A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order, Competence in Professional Ethics Holistic Technologies, Production Systems and Management Models-Typical Case Studies, Strategies for Transition towards Value-based Life and Profession	3 hours
Question paper pattern: SEE paper shall be set for 50 questions, each of the 01 mark. The pattern of the question paper is MCQ (multiple choice questions)..	

Text books:

The Textbook A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G P Bagaria, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034- 47-1

The Teacher‟s Manual for A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G

Reference Books:

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amar kantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews

BUHK408 – UHV for 2022 Scheme

4

7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj – Pandit Sunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)
14. Sussan George, 1976, How the Other Half Dies, Penguin Press. Reprinted 1986, 1991
15. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, Limits to Growth – Club of Rome’s report, Universe Books.
16. A Nagraj, 1998, Jeevan Vidya Ek Parichay, Divya Path Sansthan, Amarkantak.
17. P L Dhar, RR Gaur, 1990, Science and Humanism, Commonwealth Publishers.
18. A N Tripathy, 2003, Human Values, New Age International Publishers.
19. SubhasPalekar, 2000, How to practice Natural Farming, Pracheen (Vaidik) KrishiTantraShodh, Amravati.
20. E G Seebauer & Robert L. Berry, 2000, Fundamentals of Ethics for Scientists & Engineers ,
Oxford University Press
21. M Govindrajran, S Natrajan & V.S. Senthil Kumar, Engineering Ethics (including Human
Values), Eastern Economy Edition, Prentice Hall of India Ltd.
22. B P Banerjee, 2005, Foundations of Ethics and Management, Excel Books.
23. B L Bajpai, 2004, Indian Ethos and Modern Management, New Royal Book Co.,

Lucknow.
Reprinted 2008.

Web links and Video Lectures

Value Education websites, https://www.uhv.org.in/uhv-ii,

- http://uhv.ac.in,
- http://www.uptu.ac.in
- Story of Stuff,
  http://www.storyofstuff.com
- Al Gore, An Inconvenient Truth, Paramount Classics, USA
- Charlie Chaplin, Modern Times, United Artists, USA
- IIT Delhi, Modern Technology – the Untold Story
- Gandhi A., Right Here Right Now, Cyclewala Productions
- https://www.youtube.com/channel/UCQxWr5QB_eZUnwxSwxXEkQw
- https://fdp-si.aicte-india.org/8dayUHV_download.php
- https://www.youtube.com/watch?v=8ovkLRYXIjE

https://www.youtube.com/watch?v=OgdNx0X923I

- https://www.youtube.com/watch?v=nGRcbRpvGoU
- https://www.youtube.com/watch?v=sDxGXOgYEKM

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	They would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind.
	CO2	They would have better critical ability.
	CO3	They would also become sensitive to their commitment towards what they have understood (Hunman Values, Human relationships and Human society)
	CO4	It is hoped that thy would be able to apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction

INTERNET OF THINGS

Course Code	**22ISAE481**	Credits:01
CIE:50	SEE:50	SEE:3hrs
Hours/Week:02		Total hours:28

Prerequisite: : The students should have good knowledge of Internet of things

Course objectives:

- To make familiarize with the Iot devices
- Understanding the working of Different sensors
- Demonstrate the working of Arduino board

List of Experiments

1. Write study and installation of Arduino IDE.
2. To interface LED with arduino and write a program to 'turn on' LED for 1sec after every 2 second.
3 .To interface push button with arduino and write a program to turn on LED when push button is pushed.
4.To interface DHT11 sensor with arduino and write a program to print temperature and humidity readings.
5.To interface LDR sensor with arduino and Write a program to turn on LED when Sensor is detecting.
6.To interface DC motor with arduino and turn on the motor.
7.To interface Bluetooth with arduino and write a program to send sensor data to Smartphone using Bluetooth.
8.To interface Bluetooth with Adriano and write a program to turn On/Off when 1/0 is received for Smartphone using Bluetooth.
9.To interface buzzer with arduino and write a program to activate the buzzer with a delay of 2 second.
10.Write a program on arduino to upload temperature and humidity data to thing speak cloud.
11.To interface 7-segment display with an arduino board and develop a program to display numeric value on the display.
12.To interface arduino soil moisture sensor.
13.To interface arduino servo motor.
14.To interface ultrasonic sensor with arduino and write a program to measure the distance of an object.

Question paper pattern:

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

Reference Books: Lab Manual

Course outcomes:

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

Course Code	CO #	Course Outcome (CO)
	CO1	Demonstrate the working of Ultra sonic sensor and Temperature sensor
	CO2	Demonstrate the ability to transmit data wirelessly between different devices.
	CO3	Demonstrate the working of simple IoT task of LED control
	CO4	Implement interfacing of various sensors with Arduino
	CO5	Apply IoT concepts in advance applications

MERN

Course Code	**22ISAE482**	Credits:01
CIE:50	SEE:50	SEE:3hrs
Hours/Week:02		Total hours:28

Course objectives:

- Understand and apply critical web development languages and tools to create dynamic and responsive web applications.
- To build server-side applications using Node.js and Express
- Develop user interfaces with React.js,
- Manage data using MongoDB, and integrate these technologies to create full stack apps
- Understanding APIs and routing.

List of Experiments

1. Using MongoDB, create a collection called transactions in database usermanaged (drop if it already exists) and bulk load the data from a json file, transactions.json b. Upsert the record from the new file called transactions_upsert.json in Mongodb.
2. Query MongoDB with Conditions: [Create appropriate collection with necessary documents to answer the query] a. Find any record where Name is Somu b. Find any record where total payment amount (Payment.Total) is 600. c. Find any record where price (Transaction.price) is between 300 to 500. d. Calculate the total transaction amount by adding up Payment.Total in all records.
3. a. Write a program to check request header for cookies.
   b. write node.js program to print the a car object properties, delete the second property and get length of the object.

4 a. Read the data of a student containing usn, name, sem, year_of_admission from node js and store it in the mongodb
b.For a partial name given in node js, search all the names from mongodb student documents created in Question(a)

5.Implement all CRUD operations on a File System using Node JS

6. Develop the application that sends fruit name and price data from client side to Node.js server using Ajax

7. Develop an authentication mechanism with email_id and password using HTML and Express JS (POST method)

8. Develop two routes: find_prime_100 and find_cube_100 which prints prime numbers less than 100 and cubes less than 100 using Express JS routing mechanism

9. Develop a React code to build a simple search filter functionality to display a filtered list based on the search query entered by the user.

10. Develop a React code to collect data from rest API.

Question paper pattern:

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

Reference Books:Lab Manual

Vasan SubramanianPro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Apress; 1st ed. edition (1 April 2017) ● . Eddy Wilson Iriarte Koroliova,MERN Quick Start Guide, Packt Publishing (31 May 2018), ● https://www.geeksforgeeks.org/mern-stack/ ● https://blog.logrocket.com/mern-stack-tutorial/

Course outcomes:

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

Course Code	CO #	Course Outcome (CO)
	CO1	Apply the fundamentals of MongoDB, such as data modelling, CRUD operations, and basic queries to solve given problem.
	CO2	Use constructs of Express.js, including routing, software and constructing RESTful APIs to solve real world problems.
	CO3	Develop scalable and efficient RESTful APIs using NodeJS.
	CO4	Develop applications using React, including components, state, props, and JSX syntax.

Scala

Course Code	**22ISAE483**	Credits:01
CIE:50	SEE:50	SEE:3hrs
Hours/Week:02		Total hours:28

Course objectives:

- Model data using algebraic data types, represented in Scala as families of sealed traits and case classes.
- Use structural recursion and pattern matching to traverse and transform data.
- Learn programming with the common data structures of Scala
- Learn object-oriented programming in Scala

List of Experiments

1. a. Write a Scala program to compute the sum of the two given integer values. If the two values are the same, then return triples their sum.
   b. Write a Scala program to check two given integers, and return true if one of them is 22 or if their sum is 32.
2. a. Write a Scala program to remove the character in a given position of a given string. The given position will be in the range 0...string length -1 inclusive.
   b. Write a Scala program to create a new string taking the first 5 characters of a given string and return the string with the 5 characters added at both the front and back.
3. a. Write a Scala program to print the multiplication table of a given number using a for loop
   b. Write a Scala program to find the largest element in an array using pattern matching
4. a. Write a Scala function to calculate the product of digits in a given number
   b. Write a Scala function to check if a given number is a perfect square
5. a. Write a Scala program that creates a subclass Student that extends the Person class. Add a property called grade and implement methods to get and set it.
   b. Write a Scala program that creates a class Triangle with properties side1, side2, and side3 . Implement a method isEquilateral to check if the triangle is equilateral.
6. a. Write a Scala program that creates an enum class Color with values for different colors. Use the enum class to represent an object's color.
   b. Write a Scala program that creates a class ContactInfo with properties name, email, and address. Create a class Customer that includes a ContactInfo object.
7. a. Write a Scala program to create a set and find the difference and intersection between two sets. b. Write a Scala program to create a set and find the second largest element in the set
8. a. Write a Scala program to create a list in different ways. Note: Use Lisp style, Java style, Range list, Uniform list, Tabulate list

b.Write a Scala program to flatten a given List of Lists, nested list structure.

9 a. Write a Scala program to add each element n times to a given list of integers.

b.Write a Scala program to split a given list into two lists

10. a. Write a Scala program to swap the elements of a tuple Further print no swapping required if elements are same.

b. Write a Scala program to find non-unique elements in a tuple

Question paper pattern:

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

Reference Books:Lab Manual

Suggested Learning Resources: ● Programming Scala, Third Edition, O'Reilly Media. ● Paul Chiusano, Rúnar Bjarnason, Functional Programming in Scala 1st Edition, Manning Publications ● https://docs.scala-lang.org/tutorials/scala-for-java-programmers.html ● https://www.javatpoint.com/scala-tutorial

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Get familiar with the Scala syntax and object-oriented principles
	CO2	Learn advanced concepts - loops, expressions, inheritance, pattern matching.
	CO3	Learn to write clean and functional Scala codes and test it
	CO4	Learn functional programming using Scala

Julia		
Course Code	**22ISAE484**	Credits:01
CIE:50	SEE:50	SEE:3hrs
Hours/Week:02		Total hours:28
Prerequisite: :		

Course objectives:

- To introduce the basics of Julia programming language
- To illustrate the data structures of Julia programming language
- To make use of built-in functions and packages

List of Experiments

1 a. Develop a Julia program to simulate a calculator (for integer and real numbers.

b. Develop a Julia program to add, subtract, multiply and divide complex numbers.

c. Develop a Julia program to evaluate expressions having mixed data types (integer, real, floating -point number and complex).

2. a. Develop a Julia program for the following problem: A computer repair shop charges $100 per hour for labour plus the cost of any parts used in the repair. However, the minimum charge for any job is $150. Prompt for the number of hours worked and the cost of parts (which could be $0) and print the charge for the job.

b. Develop a Julia program to calculate a person's regular pay, overtime pay and gross pay based on the following: If hours worked is less than or equal to 40, regular pay is calculated by multiplying hours worked by rate of pay, and overtime pay is 0. If hours worked is greater than 40, regular pay is calculated by multiplying 40 by the rate of pay, and overtime pay is calculated by multiplying the hours in excess of 40 by the rate of pay by 1.5. Gross pay is calculated by adding regular pay and overtime pay

3. a. An amount of money P (for principal) is put into an account which earns interest at r% per annum. So, at the end of one year, the amount becomes P + P×r/100. This becomes the principal for the next year. Develop a Julia program to print the amount at the end of each year for the next 10 years. However, if the amount ever exceeds 2P, stop any further printing. Your program should prompt for the values of P and r.

b. Develop a Julia program which reads numbers from a file (input.txt) and finds the largest number, smallest number, count, sum and average of numbers.

4. a. Develop a Julia program and two separate functions to calculate GCD and LCM.

b. Develop a Julia program and a recursive function to calculate factorial of a number.

c. Develop a Julia program and a recursive function to generate Fibonacci series.

5. a. Develop a Julia program which reads a string (word) and prints whether the word is palindrome.

b. Develop a Julia program which reads and prints the words present in a file (input.txt) having Random Data in which words are dispersed randomly (Assumption: a word is a contiguous sequence of letters. A word is delimited by any non-letter character or end-of-line).

6. a. Develop a Julia program to determine and print the frequency with which each letter of the alphabet is used in a given line of text.

b. A survey of 10 pop artists is made. Each person votes for an artist by specifying the number of the artist (a value from 1 to 10). Develop a Julia program to read the names of the artists, followed by the votes, and find out which artist is the most popular.

7. a. Given a line of text as input, develop a Julia program to determine the frequency with which each letter of the alphabet is used (make use of dictionary)

b. Develop a Julia program to fetch words from a file with arbitrary punctuation and keep track of all the different words found (make use of set and ignore the case of the letters: e.g. to and To are treated as the same word).

8. a. Develop a Julia program to evaluate expressions consisting of rational, irrational number and floating point numbers)

b. Develop a Julia program to determine the following properties of a matrix: determinant, inverse, rank, upper & lower triangular matrix, diagonal elements, Euclidean norm and Square Root of a matrix.

9. a. Develop a Julia program to determine addition and subtraction of two matrices (element -wise).

b. Develop a Julia program to perform multiplication operation on matrices: scalar multiplication, element-wise multiplication, dot product, cross product.

10. a. Develop a Julia program to generate a plot of (solid & dotted) a function: y=x2 (use suitable data points for x).

b. Develop a Julia program to generate a plot of mathematical equation: y = sin(x) + sin(2x).

c. Develop a Julia progra

Question paper pattern:

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

Reference Books:Lab Manual

● BOOK 1: Julia – Bit by Bit (Programming for Beginners), by Noel Kalicharan, Springer: ISBN 978- 3-030-73935-5, doi: https://doi.org/10.1007/978-3-030-73936-2, 2021. ● BOOK 2: Beginning Julia Programming (For Engineers and Scientists), by Sandeep Nagar, Apress-Sp

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Apply concepts of data-types, selection and looping constructs of Julia programming language.
	CO2	Demonstrate the use of strings, functions, arrays and matrix operations in solving problems..
	CO3	Develop programs involving data structures to handle multi-valued data items.
	CO4	Make use of packages to generate plots of mathematical functions and equations.

NATIONAL SERVICE SCHEME

Course Code	**22NS49**	CIE:50
Semester:4	Credits NCMC – Non Credit Mandatory Course (Completion of the course shall be mandatory for the award of degree)	

SEE: Activities Report Evaluation by College NSS Officer at the end of every semester (3rd to 6th semester)

Course objectives:

National Service Scheme (NSS) will enable the students to:

1. Understand the community in general in which they work.
2. Identify the needs and problems of the community and involve them in problem –solving.
3. Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.
4. Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.
5. Develop capacity to meet emergencies and natural disasters & practice national integration and social harmony in general.

General Instructions - Pedagogy :

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the activities will develop students' theoretical and applied social and cultural skills.
2. State the need for NSS activities and its present relevance in the society and Provide real-life examples.
3. Support and guide the students for self-planned activities.
4. You will also be responsible for assigning homework, grading assignments and quizzes, and documenting students' progress in real activities in the field.
5. Encourage the students for group work to improve their creative and analytical skillsModules

Topics or activities to be covered

1.Water conservation techniques – Role of different stakeholders– Implementation.

2. Preparing an actionable business proposal for enhancing the village income and approach for implementation.

3.Helping local schools to achieve good results and enhance their enrolment in Higher/ technical/ vocational education

Suggested Learning material:

Books : 1. NSS Course Manual, Published by NSS Cell, VTU Belagavi.

2. Government of Karnataka, NSS cell, activities reports and its manual.

3. Government of India, nss cell, Activities reports and its manual.

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Understand the importance of his / her responsibilities towards society.
	CO2	Analyze the environmental and societal problems/issues and will be able to design solutions for the same
	CO3	Evaluate the existing system and to propose practical solutions for the same for sustainable development.
	CO4	Implement government or self-driven projects effectively in the field.
	CO5	Develop capacity to meet emergencies and natural disasters & practice national integration and social harmony in general.

PHYSICAL EDUCATION		
Course Code	**22PE49**	CIE:50
Semester:4	SEE-00	Hours/week:0-0-3
Total Hours:28	Total Marks:50	

Guideline for Athletic and Sports

Semester	Course Title	Content	No. of Hours
4th sem	Athletics Track- Sprints Jumps- Long Jump Throws- Shot Put	Track Events 1.1. Starting Techniques: Standing start and Crouch start (its variations) use of Starting Block. 1.2. Acceleration with proper running techniques. 1.3. Finishing technique: Run Through, Forward Lunging and Shoulder Shrug. Long Jump: Approach Run, Take-off, Flight in the air (Hang Style/Hitch Kick) and Landing Shot put: Holding the Shot, Placement, Initial Stance, Glide, Delivery Stance and Recovery (Perry O'Brien Technique	Total 32 hrs 2 hrs / week
	Volleyball	A. Fundamental skills 1. Service: Under arm service, Side arm service, Tennis service, Floating service. 2. Pass: Under arm pass, Over head pass. 3. Spiking and Blocking. 4. Game practice with application of Rules and Regulations B. Rules and their interpretation and duties of officials.	
	Throw ball	A. Fundamental skills: Overhand service, Side arm service, two hand catching, one hand overhead return, side arm return. B. Rules and their interpretations and duties of officials	
	Athletics Track- 110 &400 Mtrs Hurdles Jumps- High Jump Throws- Discuss Throw	110 Mtrs and 400Mtrs: Hurdling Technique :Lead leg Technique, Trail leg Technique ,Side Hurdling, Over the Hurdles Crouch start (its variations) use of Starting Block. Approach to First Hurdles, In Between Hurdles, Last Hurdles to Finishing. High jump: Approach Run, Take-off, Bar Clearance (Straddle) and Landing. Discus Throw: Holding the Discus, Initial Stance Primary Swing, Turn, Release and Recovery (Rotation in the circle).	

YOGA AND MEDITATION

Subject Code	**22YO49**	Credits:00
CIE:50	SEE: 0	SEE:--
Hours/Week: (L:T:P: S):0:0:3:0		Total hours:28

Prerequisite: NIL

Course objectives: The Course will Enable students to

- To enable the student to have good health.
- To practice mental hygiene
- To possess emotional stability.
- To integrate moral values.
- To attain higher level of consciousness.

Semester	**Content**
IV Semester	1) Patanjali's Ashtanga Yoga 2) Suryanamaskara 3) Different types of Asanas a. Sitting b. Standing c. Prone line d. Supine line 4) Kapalbhati 5) Pranayama

Guidelines

Semester	**Course Title**	**Content**	**No.of hours**

	Patanjali’s Ashtanga Yoga i)Yama ii)Niyam	Patanjali’s Ashtanga Yoga, its need and importance. Yama:Ahimsa, Satya,asteya,bramacarya,aparigraha, Niyama:Shouch, santosh, tapa, svaadhyaya,Eshwar Pranidhan	
	Surya Namaskar	Surya Namaskar 12 counts 4 rounds	**32 hours 2hrs/week**
	Different types of Asanas a. Sitting 1. Sukhasana 2.Paschimottan asana b. Standing 1. Ardhakati Chakrasana 2. Parshva Chakrasana c. Prone line 1. Dhanurasana d. Supine line 1. Halasana 2. Karna Peedasana	Asana, Need, importance of Asana. Different types of asana. Asana its meaning by name, technique, precautionary measures and benefits of each asana Different types of Asanas	
	Kapalabhati.	Meaning, importance and benefits of Kapalabhati. 40 strokes/min 3 rounds	

	Pranayama – 1. Suryanuloma – Viloma 2. Chandranuloma-Viloma 3. Suryabhedana 4. Chandra Bhedana 5. Nadishodhana	Meaning by name, technique, precautionary measures and benefits of each Pranayama	

Suggested Books

1.Yogapravesha in Kannada by Ajitkumar
2. Light on Yoga by BKS Iyengar
3. Teaching Methods for Yogic practices by Dr. M L Gharote & Dr. S K Ganguly
4. Yoga Instructor Course hand book published by SVYASA University, Bengaluru
5. Yoga for Children –step by step – by Yamini Muthanna

Course outcomes:

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

Course Code	**CO #**	**Course Outcome (CO)**
	CO1	Understand the meaning, aim and objectives of Yoga.
	CO2	Perform Suryanamaskar and able to teach its benefits.
	CO3	Understand and teach different Asanas by name, its importance, methods and benefits
	CO4	Instruct Kapalabhati and its need and importance
	CO5	Teach different types of Pranayama by its name, precautions, procedure and uses
	CO6	Coach different types of Kriyas , method to follow and usefulness