# CURRICULUM
# FOR B.E. III SEMESTER

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ESTD : 1958

## POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING
## (An autonomous college under VTU)
## KALABURAGI

## SCHEME OF TEACHING FOR III SEMESTER

### B.E. (COMPUTER SCIENCE AND ENGINEERING)

| Code No. | Course | Hours/Week | | | | Maximum Marks | | |
|---|---|---|---|---|---|---|---|---|
| | | Lecture | Tutorial | Practical | Credit | CIE | SEE | Total |
| SEMESTER III | | | | | | | | |
| 19MA31D | Computational Methods for Computer Science | 2 | 2 | 0 | 3 | 50 | 50 | 100 |
| 19CS32 | Mathematical Foundations of Computer Science | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
| 19CS33 | Object Oriented Programming | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
| 19CS34 | Data Structures | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
| 19CS35 | Analog and Digital Electronics | 3 | 2 | 0 | 4 | 50 | 50 | 100 |
| 19HU36 | Constitution of India and Professional Ethics | 2 | 0 | 0 | 0 | 50 | 50 | 100 |
| 19CSL31 | Object Oriented Programming Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
| 19CSL32 | Data Structures Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
| 19CSL33 | Analog and Digital Electronics Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
| | **Total** | **18** | **4** | **6** | **21** | **450** | **450** | **900** |

| **Course Title: COMPUTATIONAL MATHEMATICS - I** | | |
|---|---|---|
| Subject Code : **19MA31D** | Credit : 03 | CIE: 50 |
| Number of Lecture Hours/Week | **2 (L) +2(Tut)** | SEE: 50 |
| Total Number of Lecture Hours | **28** | SEE Hours: 03 |

**Prerequisites:** 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

- Interpolation methods , Numerical differentiation and Numerical integration
- Fourier Series and Z-transformation and its application in engineering fields
- Methods of least squares to fit straight line and second degree parabola
- Solve the problems using probability theory

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Finite differences: (**Forward and Backward differences), Interpolation, Newton's Forward and Backward formulae. Lagrange's interpolation and inverse interpolation formulae.<br>**Numerical differentiation:** Numerical differentiation using Newton's forward and backward interpolation formulae and problems.<br>**Numerical integration**: Trapezoidal rule, Simpsons $1/3^{rd}$ and $3/8^{th}$ rule, Weddle's rule (all formulae and rules without proof) | **6 hours** |
| **Module II**<br>**Difference equations and Z-Transforms :**<br>Difference equations –Basic definitions, Z-Transform-Definitions, standard Z-transform, linearity property , damping rule, shifting rule , initial value theorem ,final value theorem . Inverse Z-Transform and problems. | **6 hours** |
| **Module III**<br>**Fourier series:** Periodic functions, Fourier series with periods $(0, 2\pi)$, $(-\pi, \pi)$, $(0, 2l)$ and $(-l, l)$. Half range Fourier series, Practical harmonic analysis and problems. | **6 hours** |
| **Module IV**<br>**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. | **5 hours** |
| **Module V**<br>**Time Series and Forecasting:** Moving averages, smoothening of curves, | |

| | |
|---|---|
| forecasting models and methods, Statistical Quality Controls methods. **Testing of Hypothesis:** Test of significance, Chi-Square test, T-Test, ANOVA,F-Test, applications to medicine, agriculture, etc.. | **5 hours** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**
1. Higher Engineering Mathematics by B.S.Grewal, Khanna publishers; 40th Edition.2007**.**

**REFERENCES BOOKS:**
1.Advanced Engineering Mathematics by E. Kreyszig, John Willey & sons8th Edn.
2.A short course in differential equations – Rainvile E.D.9th Edition.
3.Advanced Engineering Mathematics by R.K.Jain & S.R.K Iyengar; Narosa publishing House.
4.Introductory methods of numerical analysis by S.S.Sastry
5. Probability statistics and Queuing theory- Kishore Trivedi./??check the title

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19MA31D** | **CO1** | Compute derivatives of the functions numerically using given data Computation of interpolation polynomials and numerical integration. |
| | **CO2** | Analyze discrete type system using convolution and the Z-transform. |
| | **CO3** | Construction of Fourier series for periodic signals and Fourier series to analyze circuits. |
| | **CO4** | Apply optimization techniques for real life problems |
| | **CO5** | Apply Statistical control methods and apply LPP for real-life problems in agriculture, medicine etc. |

| **Course Title: MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE** | | |
|---|---|---|
| Subject Code : **19CS32** | Credit : 03 | CIE: 50 |
| Number of Lecture Hours/Week | 03 | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** Engineering Mathematics | | |
| **Course Objectives:**<br>To develop mathematical thinking and problem-solving skills associated with writing proofs.<br>To expose students to a wide variety of mathematical concepts those are used in the Computer Science discipline. | | |
| **MODULES** | | **Teaching Hours** |
| **Module I**<br>**Mathematical logic**: Basic Connectives and truth tables, Logic Equivalence-The Laws of logic, Logical Implications-Rules of Inference.<br>**Counting:** Permutations, combination, Pigeonhole, Principles.<br>**Relations and Digraphs**: Product Sets and Partitions, Relation and Digraphs, Properties of Relations and Digraphs Properties of Relations, Equivalence Relations, Data structures for Relations and Digraphs, Operations on Relations, Transitive Closure and Warshall‟s Algorithm. | | **09 hrs** |
| **Module II**<br>**Function:** Function, Function for Computer Science, Growth of functions, Permutation Functions<br>**Order Relations and Structure:** Partially Ordered Sets, External Elements of Partially, Ordered Sets, Lattices, Finite Boolean Algebras, Functions on Boolean Algebras, Circuit. | | **08 hrs** |
| **Module III**<br>**Introduction to Graph Theory-I:** Definition & Examples, Sub-graph, complements and graph Isomorphism, Vertex degree, Euler trails and circuits.<br>**Graph Theory-II:** Planar graphs, Hamilton paths and cycles, Graph coloring, chromatic polynomials, Transport networks. (Problem solving using C) | | **08 hrs** |
| **Module IV**<br>**Trees:** Definitions, Properties, and Examples Rooted Trees, pre order traversals and post order traversals, Trees and Sorting, Weighted Trees and Prefix Codes, minimal spanning tree.<br>**Languages and finite state machines:** Languages, representations of special grammars and languages, finite state machine, semi groups machines and languages | | **08 hrs** |
| **Module V**<br>**Algebraic structures:** Semigroups, monoids, definition, example and elementary properties, Homomorphism, isomorphism and cyclic groups, cosets and lagranges theorem, elements of coding theory, the hamming matric, parity check and generator matrices, **Groups coding:** coding with coset headers and hamming matrices. Decoding in cosets: the cycle index, polays method of enumeration. | | **09 hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**

1. Grimaldi R. P., *"Discrete and Combinatorial Mathematics"*, 6th edition, Pearson Education 2004.
2. B.Kolman and R.C.Busby, "*Discrete Mathematical Structures for Computer Science*", PHI, New Delhi, 1994.

**REFERENCES:**

1. Frank Harary, *"Graph Theory"*, Addison Wesley Publishing Company, 1995.
2. C. L. Liu C. L., *"Elements of Discrete Mathematics"*, 2nd edition , McGraw Hill, Singapore, 1985.
3. J.P. Tremblay, *"Discrete Mathematical Structures with Applications to Computer Science"*, McGraw Hill, N.Y., 1977
4. Kenneth H Rosen, *"Discrete Mathematics and its applications"*, 6th Edition, McGraw Hill 2007.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS32 | CO1 | Acquire knowledge of mathematical logic, proofs of basic discrete probability, number theory and apply in problem solving |
| | CO2 | Apply various concept of functions and relations for solving computing problems |
| | CO3 | Demonstrate knowledge of fundamental concept in graphs , trees and its properties using various modeling techniques |
| | CO4 | Design grammars, finite state machines and its algebraic structures. |
| | CO5 | Discuss recurrence relations , generating functions, algebraic systems and their applications in coding theory and group theory |

| **Course Title: OBJECT ORIENTED PROGRAMMING** | | |
|---|---|---|
| **Subject Code : 19CS33** | Credit :4 | CIE: 50 |
| Number of Lecture Hours/Week | 4 | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |
| **Prerequisites: Programming in C** | | |
| **Course Objectives:**<br>• Formulate the problems in a better way giving high reliability, adaptability and extensibility to the applications.<br>• Provides a base for the further object oriented programming like Java, C# and .Net. | | |
| **MODULES** | | **Teaching Hours** |
| **Module I**<br>**Object-Oriented Paradigm**: Introduction to program paradigms, OOP's , A new paradigm, Evolution of programming paradigms, Structured versus object-oriented development, Elements of object-Oriented programming, Objects, Classes, Multiple views of the same object, Encapsulation and data abstraction, Inheritance, Delegation – Object composition, Polymorphism**.**<br>**Arrays and Strings:** Introduction, Operations on arrays, Array illustrations, Multi-dimensional arrays, Strings, Strings manipulations, Arrays of strings, Evaluation order/Undefined behaviors. | | **11 Hrs** |
| **Module II**<br>**Modular programming with functions**: Introduction, Function components, Passing data to functions, Function return data type, Library Functions, Parameter passing, Return by reference, Default arguments, Inline functions, Function overloading, Function Templates, Arrays an functions, C++ Stack, Scope and extent of variables, Storage Classes, Functions with variables number of arguments.<br>**Classes and Objects**: Introduction, Class Specification, Class Objects, Accessing class members, Defining member functions, Outside member functions as inline, Accessing member functions within the class, Data hiding, Access boundary of objects revisited, Empty classes, Pointers within a class, Passing objects as arguments, Returning objects from functions, friend functions and friend classes, Constant parameters and member functions, Structures and Classes. | | **10 Hrs.** |
| **Module III**<br>**Object Initialization:** Constructors, Parameterized constructors, Destructor, Constructor overloading, Order of construction and destruction, Constructors with default arguments, Dynamic initialization through constructors, Constructors with dynamic operations, Copy constructor, Constructors for two-dimensional arrays, Constant objects and constructor, Static data members with constructors and destructors.<br>**Operator Overloading:** Introduction, Overloadable operators, Unary operator overloading, Operator keyword, Operator return values, Binary operator overloading, Arithmetic operators, Concatenation of strings, Comparison operators, Arithmetic assignment operators, Overloading of new and delete operators, Data conversion, Overloading with friend functions, Assignment operator overloading. | | **10 Hrs.** |

| | |
|---|---|
| **Module IV**<br>**Inheritance and Virtual Functions:** Introduction, Derived class declaration, Forms of inheritance, Inheritance and member accessibility, Constructors in derived classes, Destructors in Derived classes, Abstract Classes, Multilevel inheritance, multiple inheritance, Hierarchical inheritance, Multipath inheritance and virtual base class, Hybrid inheritance.<br>**Virtual Functions**: Introduction, Need for virtual functions, Pointer to derived class objects, definition of virtual functions, Array of pointers to base class objects, pure virtual functions. | **10 Hrs.** |
| **Module V**<br>**Generic Programming with Templates**:<br>Introduction, Function templates, Overloaded Function Templates, Nesting of Function Calls, Multiple Argument Function Template, Class Templates, Inheritance of class templates, Class Templates with Overloaded Operators.<br>**Stream Computations with Console-**<br>Introduction, Predefined Console Streams, Hierarchy of Console Stream Classes, Unformatted Console I/O Operations, Formatted Console I/O Operations., Manipulators, Stream Operators with User-defined Classes<br>**Stream Computation with Files –**Introduction . Hierarchy of File Stream Classes, Opening and Closing of Files, File Modes File Pointers and their Manipulations. | .<br>**11 Hrs.** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**

1. K.R.Venugopal, Rajkumar, T . Ravishankar , " Mastering C++" , Tata McGraw-Hill Publishing Company Ltd. New Delhi.
2. E. Balagurusamy , "Object-Oriented Programming with C++", 2nd Edition , Tata McGraw-hill Publishing Company Ltd. New Delhi

**REFERENCES:**

1. Herbert Schildt, "The complete reference C++", 4th Edition, Tata McGraw-Hill Publishing Company Ltd. New Delhi, 2005. University Press, 2006.
2. Stanley B Lippman, Josee Lajoie, "C++ Primer" , Barbara E Moo, 4th Edition, Addison Wesley, 2005
3. Object-Oriented Programming with C**++,** Sourav Sahay, Oxford University Press, 2006.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS33 | CO1 | Illustrate The object oriented programming concepts and Differentiate between OOPS and conventional structured programming approaches. |
| | CO2 | Create structured, modular and re-usable code. |
| | CO3 | Demonstrate Object features and Reusability using operator overloading. |
| | CO4 | Explain Dynamic Object creation using Virtual functions. |
| | CO5 | Adopt generic programming and streams for developing Applications. |

| **Course Title: DATA STRUCTURES** | | |
|---|---|---|
| Subject Code : **19CS34** | Credits :4 | CIE: 50 |
| Number of Lecture Hours/Week | 4 Hrs | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |

**Prerequisites:** C language fundamentals and programming skill, Basic knowledge of algorithm development, Knowledge of linear and Non-linear data types

**Course Objectives:**

- To understand the behavior of data structures such as stacks, queues, trees, hash tables,search trees and their representations.
- To choose the appropriate data structure for a specified application.
- To analyze various searching and sorting algorithms.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module - I**<br>**Structures and Unions:** Structure definition, giving value to members, Structure initialization, Comparison of structure variables , Arrays of structures, Arrays within structures, Structure within structures, Structure and functions, Unions, Size of structures, Bit-fields.<br>**Pointers:** Understanding pointers, and the address of operator, Declaring and initializing pointer, Accessing a variable through it's pointer, Pointer and arrays, Pointer and character strings, Pointer and functions, Pointer and Structures.<br>**Dynamic memory allocation:** Meaning of dynamic memory allocation, MALLOC, CALLOC, Free and REALLOC functions, Pointer revisited.<br>**File management:** Definition and opening a file, closing a file, I/O operations on files, Error handling during file operation, Radom access to files, Command line arguments | **10 Hrs** |
| **Module - II**<br>**Definition and Representing Stack in C:** Primitive operation, Example. Implementing the pop() operation, Testing for exceptional conditions, Implementing the push() operation, Example: Infix, Postfix and Prefix, Basic definitions and Examples, Evaluating a postfix expression, Program to evaluate postfix expression , Converting an expression from infix to postfix, Program to convert expression from infix to postfix.<br>**Recursive definition and processes:** Factorial function, Multiplication of natural numbers, Fibonacci sequence, Binary search, Properties of recursive definition or algorithm Recursion in C: Factorial of a number Generation of Fibonacci numbers, Binary searching, Towers of Hanoi problem. | **10 Hrs** |
| **Module – III**<br>**The queue and it's sequential representation:** C implementation of queues, Insert operation, Priority queues, Array implementation of priority Linked lists: Inserting and removing nodes from a list. Linked implementation of stacks, Get node and Free node operations, Linked list implementation of queues, Linked list as a data structure, Example of list operations, Header nodes. Array implementation of list, Linked implementation of lists. Limitations of array implementation, Allocating and freeing dynamic variables, Linked list using dynamic variable, Queues as lists in C, Example of list operations in C, Non- integer and non -homogeneous lists. | **10 Hrs** |

| | |
|---|---|
| **Module - IV**<br>**Other list structures:** Circular lists, Stack as circular list, Queues as a circular list, Primitive operations on circular list, doubly linked list.<br>**Binary trees:** Operations on binary trees and applications of binary trees Binary tree representation: Node representation of binary tree, Internal and external nodes, Implicit array representation of binary trees, Choosing a binary tree representation, Binary tree traversals in C, Threaded Binary trees.<br>**Trees and their applications:** C representation of trees, Tree traversals, General expression as trees, Evaluating an expression tree, Constructing a tree. | **11 Hrs** |
| **Module - V**<br>**Sorting & Searching:** Binary tree sort, Simple insertion sort, Address calculation sort, Radix sort. Sequential searching, Searching an ordered table, Indexed sequential search, Interpolation search. **Tree searching:** Inserting into a binary search tree, Deleting from a binary search tree.<br>**Hashing:** Resolving hash clashed by open addressing, Choosing a hash function. | **11 Hrs** |

**Question paper pattern:**

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

**Text book :**

1. E. Balgurusamy , "*Programming in ANSI C*", 7 th Edition, Tata McGraw-Hill Publication, 2017.
2. Yedidyah Langsam, Moshe J. Augenstein and Aaron M. Tannenbaum, "*Data Structures Using C and C++"*, 2nd Edition, Prentice-Hall of India publication, 2005.

**Reference Books:**

1. Debasis Samanta, "Classic Data Structures", 2nd Edition, PHI, 2009.
2. Richard F. Gilberg and Behrouz A. Forouzan:, "Data Structures A Pseudocode Approach with C", Cengage Learning, 2005.
3. Robert Kruse & Bruce Leung, "Data Structures & Program Design in C", Pearson Education, 2007.
4. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", 2nd Edition, Pearson Education, 2007.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS34 | **CO1** | Apply the fundamental knowledge of pointers, dynamic memory allocation and recursion for designing data structures. |
| | **CO2** | Demonstrate the usage of stack, queue data structure for design of applications. |
| | **CO3** | Illustrate basic operations on linked lists and construct various data structures using linked lists. |
| | **CO4** | Design Binary trees and binary search trees using tree data structure. |
| | **CO5** | Compare, analyze and implement different sorting and searching Techniques. |

| **Course Title: Analog and Digital Electronics** | | |
|---|---|---|
| Subject Code : **19CS35** | Credits :4 | CIE: 50 |
| Number of Lecture Hours/Week | 3 Hrs (L) +2 (Tut) | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |

**Prerequisites:**

- Knowledge of Basic Electronics and Boolean algebra.

**Course Objectives:**

- Recall and Recognize characteristics of PDs, optocouplers, BJT.
- Demonstrate and analyze operational amplifier circuits and their applications
- Describe and analyze combinational logic circuits, simplifications of algebraic equations using Karnaugh maps and Quine McClaskey techniques
- Design decoders, encoders and substractors , Binary comparators latches and flip flops.
- Design registers and counters , A/D and D/A converter.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module - I**<br>Photo diodes , Light emitting diodes, Optocouplers, BJT Biasing: Fixed Bias , Collector to Base Bias, Voltage Divider Bias, Operational Amplifiers, Application circuits: Multivibrators using 555 IC, Peak detector, Schmitt trigger, Active filters, Non linear Amplifier, Relaxation Oscillator, Current to voltage, voltage to Current converter, Regulated power supply parameters, Adjustable voltage Regulator. | **10 Hrs** |
| **Module - II**<br>The Basic Gates: Review of Basic Logic gates, Positive and Negative Logic, . Combinational Logic Circuits: Sum-of-Products Method, Truth Table to Karnaugh Map, Pairs Quads, and Octets, Karnaugh Simplifications, Don't-care Conditions, Product-of-sums Method, Product-of-sums simplifications, Simplification by Quine-McClusky Method, Hazards and Hazard covers | **8 Hrs** |
| **Module – III**<br>Data-Processing Circuits: Multiplexers, Demultiplexers, 1-of-16 Decoder, BCD to Decimal Decoders, Seven Segment Decoders, Encoders, Exclusive-OR Gates, Parity Generators and Checkers, Magnitude Comparator, Programmable Array Logic, Programmable Logic Arrays,<br>Arithmetic Building Blocks: Half-adder, Full adder, Adder &Substractor, arithmetic logic unit.<br>Flip- Flops: RS Flip-Flops, Gated Flip-Flops, Edge-triggered RS FLIP-FLOP, Edge-triggered D FLIP-FLOP,T FLIP-FLOP, Edge-triggered JK FLIP-FLOP. | **8 Hrs** |

| | |
|---|---|
| **Module - IV**<br>Flip- Flops: FLIP-FLOP Timing, JK Master-slave FLIP-FLOP, Switch Contact Bounce Circuits, Various Representation of FLIP-FLOPs. Registers: Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers. Counters: Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus. | **8 Hrs** |
| **Module - V**<br>Counters: Decade Counters, Presettable Counters, Counter Design as a Synthesis problem, A Digital Clock. D/A Conversion and A/D Conversion: Variable, Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution, A/D Converter-Simultaneous Conversion, A/D Converter-Counter Method, A/D Accuracy and Resolution. | **8 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books:**

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

**Reference Books:**

1. R D Sudhaker Samuel, "Illustrative Approach to Logic Design", Sanguine-Pearson, 2010.
2. M Morris Man, "Digital Logic and Computer Design", 10th Edition, Pearson, 2008.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS35 | CO1 | Use of various devices like CRO, function generator, multimeter, bread board, Make us of electronic components, ICs, instruments and tools for design and testing of circuits for given inputs. |
| | CO2 | Evaluate and design the combinational circuit. |
| | CO3 | Evaluate and design registers and counters using flip-flops. |
| | CO4 | Design and develop D/A convertors. |
| | CO5 | Analyze the working and implementation of ALU. |

| **Course Title: CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS** | | |
|---|---|---|
| Subject Code : **19HU36** | Credit : **0** | CIE: 50 |
| Number of Lecture Hours/Week | **2 hrs** | SEE: 50 |
| Total Number of Lecture Hours | **28** | SEE Hours: 02 |

**Pre-requisites:**

**Course objectives:**

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

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

| **Modules** | **Teaching Hours** |
|---|---|
| **Module-I**<br>**Introduction and Fundamental Rights:** The Constitution of India. Evolution of the Constitution. The Constituent Assembly of India. Sources and Features of the Indian Constitution. Preamble to the Constitution of India. Salient Features of Fundamental Rights and their classification. General exercise of Fundamental Rights and their limitations. RTI (Right to Information Act of 2005 Under Article 19(1)) and The Right of Children to Free and Compulsory Education Act or Right to Education Act (RTE) Under Article 21-A of the Constitution. Article 371(J) of the Constitution applicable to Hyderabad Karnataka Area. | **06 Hrs** |
| **Module – II**<br>**Directive Principles of the State Policy and The State Executive**: Under Article 36 to 51 of The Constitution and their Relevance. Fundamental Duties Under Article 51A of The Constitution and their Relevance. State Government - The Governor- Appointment, Powers and Functions of the Governor. The Appointment of Chief Minster, his Powers and Functions. The State Council of Ministers and their Functions. The State legislature and The State Council. The High Court of the State, its Powers and Jurisdiction. Appointment and Qualifications of High Court Judges. | **06 Hrs** |

| | |
|---|---|
| **Module – III**<br><br>**The Union Executive**: Central Government. The President of India, his Election, Powers and Functions. The Vice-President of India, his Election, Powers and Functions. The Supreme Court of India and its Structure. Appointment and Qualification of Supreme Court Judges. Their Powers and Functions. The Structure of Judiciary in India. The Parliament of India. The Prime Minister, his Appointment, Powers and Functions. The Union Council of Ministers their Powers and Responsibilities. Concept of Public Interest Litigation (PIL). | **06 Hrs** |
| **Module – IV**<br><br>**Constitutional Provisions and Emergency Provisions and Election Process** : Constitutional for Women, Children, Backward Classes and Scheduled Caste and Scheduled Tribes under different Article of The Constitution. Different types of Emergencies under Article 352, 356 and 360 of the Constitution of India. The Election Commission of India- its Powers and Functions. The State Election Commission | **05 Hrs** |
| **Module – V**<br><br>**Engineering Ethics:** Its Aims and Scope, Responsibilities of Engineers, Impediments to their Responsibilities, Honesty, Integrity, Reliability, Risk and Safety Measures, Liabilities of Engineers. | **05 Hrs** |

**Question paper pattern:**
Solve five full questions selecting at least one question from each Module.

**Text books:**

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

**Reference Books:**

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

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19HU36 | **CO1** | Explain the evolution and features of constitution, fundamental rights and their classification |
| | **CO2** | Describe the directive principles of state policy, fundamental duties andThe State Executive |
| | **CO3** | Describe about The Union Executive and concept of Public Interest Litigation |
| | **CO4** | Explain the Constitutional Provisions for women, children, SC/ST'S, Emergency Provisions and Election Process |
| | **CO5** | Identifies the qualities required for an professional engineers to be ethical |

| **Course Title: OBJECT ORIENTED PROGRAMMING LAB** | | |
|---|---|---|
| Subject Code :**19CSL31** | Credit :1 | CIE: 50 |
| Number of Practical Hours | **2Hrs (Practical)** | SEE: 50 |
| Total Number of Lecture./Practical Hours | | SEE Hours: 03 |

**Prerequisites: Programming in C**

**Course Objectives:**

1. Handson experience of Object Oriented Programming Concepts with C++.
2. Preparedness to study independent layer Object Oriented Programming language and apply to variety of real time problem scenarios.

**List of Programs**

**PART – A**

1. Write a C++ program to display youngest and eldest person, age from a given set of N number of person(array).
2. Write a C++ program to find Addition and Subtraction of two M x N matrices.
3. Write a C++ program to concatenate two strings.
4. Write a C++ program to find factorial of a number using recursive function.
5. Write a C ++ program to store numbers into an array and find sum of all the elements of the array using pointers.

**PART – B**

1. Create a simple class STUDENT containing the data members, rollno, name, age & display the contents using user defined functions. Test the program with and without scope resolution operator.

2. Create a class called, EMPLOYEE containing data members, Empno, Empname, Designation, Basic pay, DA, HRA, Insurance, TAX. Develop the member functions for calculating and displaying the Netsalary (Netsalary=(Basicpay + DA+HRA)-(Insurance + TAX).

3. Write a C++ program to create class DATE. Display age of the person by considering date of birth and current date using inline function.

4. Write a C++ program to create a class ACC with data members, accno, balance. Create objects, namely,ACC1 and ACC2.Writeamemberfunction to transfer amount fromACC1toACC2. Display the new balance in the transacted accounts.
5. Create a class FIXED-DEPOSIT with data members, principal- amt, year, and rate of interest. Using constructors, initialize the principal-amt and rate of interest. Find the gross amount after a given period of investment.

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

7. Write a C++ program to create a class called STACK using an array of integers. Implement the following operations by overloading +&- and display the contents of stack.
   s1=s1 + element; where s1 is 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.

8. Write a C++ program to create a class NAME and implement the following operations. Display the result after every operation by overloading the <<.
   i) NAME firstname = "Herbert"
   ii) NAME lastname = "Schield"
   iii) NAME fullname = firstname + lastname
   (Use copy constructor)

9. 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 o two matrices m1 and m2 to be added and subtracted. Perform the addition and subtraction by overloading the operators + and –respectively. Display the results.
   if(m1==m2)
   {
   m3=m1+m2;
   m4=m1–m2;
   } else display Error message.

10. Write a C++ program to demonstrate the function overloading by overloading the user defined function called ADD with variable type and number of arguments.
    (Eg. ADD (Int,int), ADD(int, float), ADD(init,int, float) etc.)

11. Create three classes, namely, STUDENT ,EXAM and RESULT. The STUDENT class has data members, Rollno, Name and Branch and the class EXAM inherits the STUDENT class with new own data members, marks scored in six subjects. Derive the RESULT class from the EXAM class and it has its own data members, total_marks.

12. Write an inter active program to model this multilevel inheritance relationship.

13. Create a base class RESERVATION. Create the derived classes, namely, ADULT, SENIOR_CITIZEN, CHILD by multiple inheritance. The RESERVATION class has data members, Name_of_passenger, Age, Date_of_journey, Source, Destination, Ticket_charge. Write an inter active program to display the ticket charges depending upon the category of passenger.
    (Note: Charge for Children =½ of adult ticket charge. Senior citizen=¼ of adult ticket charge.)

14. Write a C++ program to create a pure virtual function and show its access from the object of a derived class through the pointer of base class.

15. Write a C++ program to sort a given array of N numbers using template function.

Note: For SEE, students will be asked to do similar programs from PART-B only..

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL31 | CO1 | Apply OOP Concepts to develop Programs |
| | CO2 | Demonstrate design of applications using Class and Objects |
| | CO3 | Adopt Inheritance feature to develop programs. |
| | CO4 | Illustrate Reusability using Function and Operator overloading |
| | CO5 | Create dynamic Objects using virtualization |

| **Course Title:** | **DATA STRUCTURES LAB** | |
|---|---|---|
| Subject Code : **19CSL32** | Credits : 1 | CIE: 50 |
| Number of Lecture Hours/Week | **2 Hrs (Practical)** | SEE: 50 |
| | | SEE Hours: 03 |

**Prerequisite:** C Language : Functions and Pointers

**Course Objectives :**

1. To study the working of data structures such as stacks, queues, trees, hash tables, search trees.
2. To choose the appropriate data structure for a specified application.
3. To learn various searching and sorting algorithms.

**List of Programs**

1. Design, Develop and Implement a menu driven Program in C for thefollowing Array operations
   a. Creating an Array of N Integer Elements
   b. Display of Array Elements with Suitable Headings
   c. Inserting an Element (ELEM) at a given valid Position (POS)
   d. Deleting an Element at a given valid Position(POS)
   e. Exit.

   Support the program with functions for each of the above operation.

2. Design, Develop and Implement a program in C for thefollowing operations on Strings
   a. Read a Main String (STR), a Pattern String (PAT) and aReplace String (REP).
   b. Perform Pattern Matching Operation: Find and Replace all occurrences of PAT in STR with REP if PAT exists in STR. Repost suitable messages in case PAT does not exist in STR.

   Support the program with functions for each of the above operations. Don"tuse built-in functions.

3. 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)
   a. Push an Element on to Stack
   b. Pop an Element from Stack
   c. Display the status of Stack
   d. Demonstrate Overflow and Underflow situations on Stack
   e. Exit

   Support the program with appropriate functions for each of the aboveoperations.

4. Design, Develop and Implement a Program in C for converting an Infix Expression to Postfix Expression. Program should support forboth parenthesized and free parenthesized expressions with the operators: +, -, *, /, %( Remainder), ^ (Power) and alphanumeric operands.

5. Design, Develop and Implement a Program in C for the followingStack Applications
   a. Evaluation of Suffix expression with single digit operandsand operators: +, -, *, /, %, ^
   b. Solving Tower of Hanoi problem with n disks

6. 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)
   a. Insert an Element on to QUEUE
   b. Delete an Element from QUEUE
   c. Demonstrate Overflow and Underflow situations on QUEUE
   d. Display the status of QUEUE
   e. Exit
   Support the program with appropriate functions for each of the aboveoperations.

7. Design, Develop and Implement a menu driven Program in C for thefollowing operations on Singly Linked List (SLL) of integer values
   a. Create a SLL of N integers by using front insertion.
   b. Display the status of SLL and count the number of nodes in it
   c. Perform Insertion and Deletion at End of SLL
   d. Perform Insertion and Deletion at Front of SLL

8. Design, Develop and Implement Program in C to Reverse a SinglyLinked List (SSL) of a given integers.

9. Design, Develop and Implement a menu driven Program in C for thefollowing operations on Priority Queue.
   a. Create a Priority queue by using Insert function.
   b. Insertion data and Priority values as Input.
   c. Perform Deletion operation.
   d. Display the elements of Priority queue.

10. Design, Develop and Implement a Program in C for the followingoperations on Binary Search Tree(BST) of Integers
    a. Create a BST of N integers: 6,9,5,2,8,15,24,14,7,8,5,2.
    b. Traverse the BST in Inorder
    c. Traverse the BST in Preorder
    e. Traverse the BST in Postorder.

11. Given a File of N employee records with a set K of Keys(4- digit) which uniquely determine the records in file F. Assume that file F is maintainedin memory by a Hash Table(HT) of m memory locations with L as the set of memory addresses (2- digit) of locations in HT. Let the keys in K and Addresses in L are Integers. Design and develop a Program in C thatuses Hash function H: K ®L as H(K)=K mod m (remainder method), and implement hashing technique to map a given key K to the address space L. Resolve the collision (if any) using linear probing.

| **Course outcomes:**<br>**On completion of the course, the student will have the ability to:** | | |
|---|---|---|
| **Course Code** | **CO #** | **Course Outcome (CO)** |
| **19CSL32** | **CO1** | Design and develop various data structure using pointers , dynamic memory allocation and recursion |
| | **CO2** | Demonstrate basic operations on linked list using suitable data structures. |
| | **CO3** | Illustrate the implementation of different sorting and searching techniques. |
| | **CO4** | Construct Binary trees and binary search trees |
| | **CO5** | Write a well organized laboratory report presenting the results in a clear way using algorithms and obtained output. |

| **Course Title: ANALOG AND DIGITAL ELECTRONICS LAB** | | |
|---|---|---|
| Subject Code : **19CSL33** | Credits : 1 | CIE: 50 |
| Number of Lecture Hours/Week | **2Hrs (Practical)** | SEE: 50 |
| | | SEE Hours: 03 |

**Prerequisite:** Knowledge of Basic Electronics and Boolean algebra.

**Course Objectives :**

- To illustrate the students different electronic circuit and their application in practice.
- To impart knowledge on assessing performance of electronic circuit through monitoring of sensitive parameters.
- To evaluate the use of computer- based analysis tools to review performance of semiconductor device circuit

**Laboratory Experiments:**

1. Design and construct a Schmitt trigger using Op-Amp for given UTPand LTP values and demonstrate its working.
2. Design and construct a rectangular waveform generator (Op-Amp relaxation oscillator) for given frequency and demonstrate its working.
3. Design and implement an Astablemultivibrator circuit using 555 timerfor a given frequency and duty cycle.
4. Design and implement Half adder, Full Adder, Half Subtractor, FullSubtractor using basic gates.
5. Given a 4-variable logic expression, simplify it using Entered VariableMap and realize the simplified logic expression using 8:1 multiplexer IC.
6. Design and implement code converter I)Binary to Gray (II) Gray toBinary Code using basic gates.
7. Design and verify the Truth Table of 3-bit Parity Generator and 4-bitParity Checker using basic Logic Gates with an even parity bit.
8. Realize a D,T,JK Flip-Flop using NAND gates and verify its truth table.
9. Design and implement a mod-n (n<8) synchronous up counter using JKFlip Flop ICs and Demonstrate its working
10. Design and implement an Asynchronous counter using decade counterIC to count from 0 to n(n<=9) and demonstrate on seven segment display(using IC 7447)
11. Generate a Ramp output waveform using DAC0800 (Inputs are given toDAC through IC74393 dual 4-bit binary counter).

**12.** To study 4-bitALU using IC-74181.

**Question paper pattern:**
**Note:** Conduction of Practical Examination: All laboratory experiments (1 to 11 nos) are to be included for practical examination.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL33 | CO1 | Use of various devices like CRO, function generator, multimeter, bread board, Make us of electronic components, ICs, instruments and tools for design and testing of circuits for given inputs. |
| | CO2 | Evaluate and design the combinational circuit. |
| | CO3 | Evaluate and design registers and counters using flip-flops. |
| | CO4 | Design and develop D/A convertors. |
| | CO5 | Analyze the working and implementation of ALU. |

# CURRICULUM

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## B.E. IV SEMESTER

## POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING
## (An autonomous college under VTU)
## KALABURAGI

**About the institution:** The Hyderabad Karnataka Education (HKE) society founded by Late Shri Mahadevappa Rampure, a great visionary and educationist.The HKE Society runs 46 educational institutions. Poojya Doddappa Appa College of Engineering, Gulbarga is the first institution established by the society in 1958. The college is celebrating its golden jubilee year, setting new standards in the field of education and achieving greater heights. The college was started with 50% central assistance and 50% state assistance, and a desire to impart quality technical education to this part of Karnataka State. The initial intake was 120 with degree offered in three branches of engineering viz, Civil, Mechanical and Electrical Engineering. Now, it houses 11 undergraduate courses, 10 post Graduate courses and 12 Research centers, established in Civil Engg., Electronics & Communication Engg, Industrial & Production Engg, Mechanical Engg, Electrical Engg., Ceramic Cement Tech., Information Science & Engg.,Instrumentation Technology, Automobile Engg., Computer Sc. and Engg., Mathematics and Chemistry All the courses are affiliated to Visveswaraya Technological University, Belgaum. At present the total intake at UG level is 980 and PG level 193.

The college receives grant in aid funds from state government. A number of projects have been approved by MHRD /AICTE, Govt. of India for modernization of laboratories. KSCST, Govt. ofKarnataka is providing financial assistance regularly for the student's projects.

The National Board of Accreditation, New Delhi, has accredited the College in the year 2005-08 for 09 UG Courses out of which 08 courses are accredited for three years and 01 course is accredited for five years. And second time accredited for Six Course in the year 2009-2012

Our college is one among the 14 colleges selected under TEQIP, sponsored by World Bank. It has received a grant of Rs 10.454 Crores under this scheme for its development. The institutionis selected for TEQIP phase II in year 2011 for four years. Institution is receiving a grant of Rs
12.50 Crores under TEQIP Phase -II scheme for its development and selected for TEQIP-III as mentoring Institute for BIET Jhansi(UP).

Recognizing the excellent facilities, faculty, progressive outlook, high academic standards and record performance, the VTU Belgaum reposed abundant confidence in the capabilities of the College and the College was conferred Autonomous Status from the academic year 2007-08, to update its own programme and curriculum, to devise and conduct examinations, and to evaluate student's performance based on a system of continuous assessment. The academic programmers are designed and updated by a Board of Studies at the department level and Academic Council at the college level. These statutory bodies are constituted as per the guidelines of the VTU Belgaum. A separate examination section headed by a Controller of Examinations conducts the examinations.

At present the college has acquired the Academic autonomous status for both PG and UG courses from the academic year 2007-08 and it is one among the six colleges in the state of Karnataka to have autonomous status for both UG and PG courses.

One of the unique features of our college is, it is the first college in Karnataka State to start the Electronics and Communication Engineering branch way back in the year 1967, to join NIT Surathkal and IISc, Bangalore. Also, it is the only college in the state and one among the three colleges across the country, offering a course in Ceramic and Cement Technology. This is the outcome of understanding by faculty and management about the basic need of this region, keeping in view of the available raw material and existing Cement Industries.

Bharatiya Vidya Bhavan National Award for an Engineering College having Best Overall Performance for the year 2017 by ISTE (Indian Society for Technical Education). In the year 2000, the college was awarded as Best College of the year by KSCST, Bangalore in the state level students projects exhibition.

The college campus is spread over 71 acres of land on either side of Mumbai-Chennai railway track and has a sprawling complex with gardens and greenery all around.

**About the department:** The Computer Science and Engineering department was started in the year 1984 with an intake of 40 students for UG. The department has seen phenomenal growthand now the department has increased UG intake to 120 students and offering two Post Graduation programmes : PG (Computer Science and Engineering with an intake of 25 students) and PG(Computer Network and Engineering with an intake of 18 students). The department is offering research program under its recognized research center. The department is having state- of-the-art computing facilities with high speed internet facilities and laboratories. The department library provides useful resources like books and journals. The department has well qualified and experienced teaching faculty. The department has been conducting several faculty development programs and student training programs.

**Vision of the institute:**

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

**Mission of the institute:**

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

**Department Vision**

To become pioneers in computer education and research and to prepare highly competentIT professionals to serve Industry and Society at local and global levels.

**Department Mission**

- To impart high quality professional education to become a leader in Computer Scienceand Engineering.
- To achieve excellence in Research for contributing to the development of the society.
- To inculcate professional and ethical behavior to serve the Industry.

**Program Educational Objectives (PEOs) of the department**

**PEO1**: To prepare graduates with core competencies in mathematical and engineering fundamentals to solve and analyze computer science and engineering problems.

**PEO2**: To adapt to evolving technologies and tools for serving the Society.

**PEO3**: To perform as team leader, effective communicator and socially responsible computerprofessional in multidisciplinary fields following ethical values.

**PEO4**: To encourage students to pursue higher studies and engage in research andentrepreneurship.

**Program Specific Outcomes (PSOs) of the department**

**PSO1**: Acquire competency in hardware and software working principles to analyze and solvecomputing problems.

**PSO2**: Design quality software to develop scientific and business applications following Software Engineering practices.

**PSO3**: Apply cutting edge technologies using modern tools to find novel solutions ethically toexisting problems.

**Program Outcomes**

Engineering Graduates will be able to:

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

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

3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

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

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

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

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

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

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

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

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

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

## SCHEME OF TEACHING FOR IV SEMESTER -2021-2022

| CodeNo. | Course | Hours/Week | | | | Maximum Marks | | |
|---|---|---|---|---|---|---|---|---|
| | | Lecture | Tutorialal | Pratecal | Credit | CIE | SEE | Total |
| **SEMESTER IV** | | | | | | | | |
| 19MA41D | Applied Statistics | 2 | 2 | 0 | 3 | 50 | 50 | 100 |
| 19CS42 | Operating System | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
| 19CS43 | Java Programming | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
| 19CS44 | Analysis & Design of Algorithm | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
| 19CS45 | Microprocessors and Microcontrollers | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
| 19CV46 | Environment Studies | 2 | 0 | 0 | 0 | 50 | 50 | 100 |
| 19KAN47/ 19KAK47 | Kannada | 1 | 0 | 0 | 1 | 50 | 50 | 100 |
| 19CSL41 | Analysis & Design of Algorithm Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
| 19CSL42 | Microprocessors and Microcontroller Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
| 19CSL43 | Java Programming Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
| | Total | **21** | **2** | **6** | **23** | **500** | **500** | **1000** |

| **Course Title: Applied Statistics** | | |
|---|---|---|
| Subject Code : **19MA41D** | Credit : 03 | CIE: 50 |
| Number of Lecture Hours/Week | **2 (Theory) + 2 (Tut)** | SEE: 50 |
| Total Number of Lecture Hours | 28 | SEE Hours: 03 |
| **Prerequisites:** Basic knowledge of Statistic and Probability | | |
| **Course Objectives:** To enable the students to obtain the knowledge of Engineering Mathematics in the following topics<br>1. Probability distribution of discrete and continuous random variables<br>2. Joint probability distributions and discrete and continuous random variables and Morkov's chains<br>3. Analyse the sample data using Large sample test, t-distribution and chi- distribution | | |

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Probability distributions:**<br>Random variable (Discrete and continuous) p.d.f., c.d.f., Binomial distribution, Poisson distributions, Normal distribution and problems | **6 hours** |
| **Module II**<br>**Joint probability distributions:** Concept of joint probability distribution, discrete and continuous random variables independent random variables .problems on expectation and variance | **6 hours** |
| **Module III**<br>**Markov chains:** Introduction probability vectors stochastic matrices, higher transition probability. Stationary distribution of regular Markov chains and absorbing<br>states | **5 hours** |
| **Module IV**<br>**Sampling theory:** Sampling, sampling distribution, standard error. Testing of hypothesis for means. Confidence limits for means. Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations. 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<br>attributes. And problems**.** | **6 hours** |
| **Module V**<br>**Distances in Classification:** Introduction, Euclidean Distance, Manhattan Distance, Euclidean vs Manhattan Distance, Chebyshev Distance, Hamming Distance, Distancecalculation in Clusters | **5 hours** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**

1 Higher Engineering Mathematics by B.S.Grewal, 36th Edn.
2 Engineering Mathematics by N. P. Bali and Manish Goyal. Laxmi publications, latest edition.
3 Higher Engineering Mathematics by H. K. Dass and Er. Rajnish Verma. S. Chand publishing 1st edition -2011

**REFERENCES:**

1. Advanced Engineering Mathematics by E. Kreyszig, John Willey & sons 8th Edn.
2. Advanced Engineering Mathematics by R.K.Jain & S.R.K Iyengar; Narosa publishing House.
3. Introductory methods of numerical analysis

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19MA41D | CO1 | Solve problems using theoretical probability distributions |
| | CO2 | Apply the concepts of joint probability, to find covariance, correlation, independent variables |
| | CO3 | Apply stochastic to find the probability vectors, stochastic matrices and higher transition probability |
| | CO4 | Analyse the sample data using Large sample tests |
| | CO5 | Analyse the sample data using t-distribution and chi- distribution . |

| **Course Title: OPERATING SYSTEM** | | |
|---|---|---|
| Subject Code : **19CS42** | Credit : 4 | CIE: 50 |
| Number of Lecture Hours/Week | **4 hrs (Theory)** | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |
| **Prerequisites: Computer Organization, Microprocessor** | | |

**Course Objectives:**

- Learn services provided by the operating system and design of operating system
- Gain knowledge on how processes are synchronized and scheduled how different resources are managed.
- Understand structure and organization of file system and approaches to memory management.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module – I**<br>**Introduction:** What Operating Systems Do, Computer-System Organization, Computer-System Architecture, Operating-System Operations, Process Management, Memory Management, Storage Management, Security and Protection, Kernel Data Structures, Computing Environments.<br>**Operating-System Structures:** Operating-System Services, User and Operating-System Interface, System Calls, Types of System Calls, System Programs, Operating-System Design and Implementation, Operating System Structure.<br>**Case Studies:** Architecture of UNIX, The Kernel of Unix; The Kernel of Solaris; Architecture of Windows. | **10 Hrs** |
| **Module-II**<br>**Process Management**: Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication, Communication in Client–Server Systems.<br>**Multithreaded Programming:** Overview, Multicore Programming, Multithreading Models, Thread Libraries, Implicit Threading, Threading Issues | **10 Hrs** |
| **Module – III**<br>**Process Scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Thread Scheduling, Multi-Processor Scheduling, Real-Time CPU Scheduling, Exercises<br>**Process Synchronization:** The Critical-Section Problem, Peterson's Solution, Synchronization hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors . | **11 Hrs** |
| **Module – IV**<br>**Deadlocks:** System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock, Exercises.<br>**Memory Management:** Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table, Example: Intel 32- and 64-bit Architectures, Example: ARM Architecture, Exercises. | **10 Hrs** |

| **Module – V** | **11 Hrs** |
|---|---|
| **Virtual Memory:** Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing, Memory mapped files, Allocating Kernel Memory, Exercises.<br><br>**File System: File-System Interface:** File Concept, Access Methods, Directory and disk Structure, File system Mounting, File Sharing, and Protection | |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text book :**

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Concepts, 9th Edition, Wiley-India, 2018.
2. D.M Dhamdhere, Operating systems - A concept based Approach, 3rd Edition, Tata McGraw- Hill, 2012.

**Reference Books:**

1. P.C.P. Bhatt: Operating Systems, 2nd Edition, PHI, 2006.
2. Harvey M Deital: Operating systems, 3rd Edition, Addison Wesley, 2003.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS42 | **CO1** | Describe the functions of operating systems and its structures |
| | **CO2** | Describe process concepts and management models. |
| | **CO3** | Apply Scheduling algorithms and different concurrency control techniques to provide co-ordination among processes for the global data. |
| | **CO4** | Apply deadlock detection and prevention algorithms and memory management including the concept of paging, segmentation and swapping policies. |
| | **CO5** | Discuss Virtual memory management and describe file system interface. |

| **Course Title: Java Programming** | | |
|---|---|---|
| Subject Code : **19CS43** | **Credit : 04** | CIE: 50 |
| Number of Lecture Hours/Week | 04 | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |
| **Prerequisites:** Concepts of Object oriented programming | | |
| **Course Objectives:** Learn the Java Programming to develop applications, creating GUI with applets, swings and web applications using servlets. | | |

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**An Overview of Java** -Object-Oriented Programming, Two Paradigms Abstraction, The Three OOP Principles, A First Simple Program , Entering the Program, Compiling the Program ,A Closer Look at the First Sample Program , A Second Short Program, The Java Keywords , The Java Class Libraries,<br>**Data Types, Variables, and Arrays:** The Primitive Types - Integers, byte, short, int, long, Floating-Point Types, float, double, Characters , Booleans. A Closer Look at Literals - Integer Literals, Floating-Point Literals, Boolean Literals , Character Literals, String Literals, Variables, Declaring a Variable, Dynamic Initialization ,The Scope and Lifetime of Variables, Type Conversion and Casting, Java's Automatic Conversions, Casting Incompatible Types, Automatic Type Promotion in Expressions, The Type Promotion Rules , Arrays, One-Dimensional Arrays, Multidimensional Arrays, Alternative Array Declaration Syntax, A Few Words About Strings. Overview of Operators | **11 Hrs** |
| **Module II**<br>**Over view of Control Statements, Introducing Classes**-Class Fundamentals, The General Form of a Class, A Simple Class, Declaring Objects, A Closer Look at new, Assigning Object Reference Variables.<br>**Introducing Methods**- Adding a Method to the Box Class, Returning a Value, Adding a Method That Takes Parameters, Constructors, Parameterized Constructors, The this Keyword, Instance Variable Hiding, Garbage Collection, The finalize( ) Method, A Stack Class<br>**Inheritance**- Inheritance Basics, Member Access and Inheritance, A More Practical Example, A Superclass, Variable Can Reference a Subclass Object.<br>Using super: Using super to Call Superclass Constructors, A Second Use for super, Creating a Multilevel Hierarchy, When Constructors Are Called, Method Overriding. | **10 Hrs** |
| **Module III**<br>**Dynamic Method Dispatch.** Why Overridden Methods? Applying Method Overriding, Using Abstract Classes, Using final with Inheritance, Using final to Prevent Overriding, Using final to Prevent Inheritance, The Object Class<br>**Packages and Interfaces:** Packages, Defining a Package, Finding Packages and CLASSPATH, A Short Package Example<br>**Access Protection:** An Access Example, Importing Packages,<br>Interfaces: Defining an Interface, Implementing Interfaces, Nested Interfaces, Applying Interfaces, Variables in Interfaces, Interfaces Can Be Extended. | |

| | |
|---|---|
| **Exception Handling:** Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Displaying a Description of an Exception, Multiple catch Clauses, Nested try Statements, throw, throws ,finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions, Using Exceptions. | **10 Hrs** |
| **Module IV**<br>**The Applet Class**-Two Types of Applets<br>**Applet Basics:** The Applet Class, Applet Architecture,<br>**An Applet Skeleton**: Applet Initialization and Termination, Overriding update( ), Simple Applet Display Methods,.<br>**Requesting Repainting:** A Simple Banner Applet, Using the Status Window , The HTML APPLET Tag.<br>**Passing Parameters to Applets:** Improving the Banner Applet, getDocumentBase( ) and getCodeBase( ), AppletContext and showDocument( ), The AudioClip Interface,The AppletStub Interface , Outputting to the Console.<br>**Event Handling**- Two Event Handling Mechanisms.<br>**The Delegation Event Model- Events:** Event Sources, Event Listeners.<br>**Event Classes:** The ActionEvent Class, The AdjustmentEvent Class, The ComponentEvent Class, The ContainerEvent Class,The FocusEvent Class,The InputEvent Class, The ItemEvent Class, The KeyEvent Class, The MouseEvent Class, The MouseWheelEvent Class, The TextEvent Class,The WindowEvent Class.<br>Using the Delegation Event Model- Handling Mouse Events, Handling Keyboard Events, Adapter Classes, Inner Classes, Anonymous Inner Classes.<br>**JDBC-ODBC Connectivity:** Talking to Database, Immediate Solutions, Essential JDBC program, using prepared Statement Object, Interactive SQL tool. JDBC in Action Result sets, Batch updates, Mapping, Basic JDBC data types, Advanced JDBC data types, immediate solutions. | **10 Hrs** |
| **Module V**<br>**Servlets -** Background, The Life Cycle of a Servlet, Using Tomcat for Servlet Development,.<br>**A Simple Servlet:** Create and Compile the Servlet Source Code , Start Tomcat , Start a Web Browser and Request the Servlet, The Servlet API,<br>**The javax servlet Package:** The Servlet Interface, The Servlet Config Interface, The Servlet Context Interface, The ServletRequest Interface, The Servlet Response Interface,The GenericServlet Class, The Servlet Input Stream Class, The ServletOutputStream Class, The Servlet Exception Classes, Reading Servlet Parameters.<br>**The javax,servlet.http Package:** The Http Servlet Request Interface,The Http Servlet Response Interface, The Http Session Interface, The Http Session Binding Listener Interface, The Cookie Class, Contents, The Http Servlet Class, The Http Session Event Class, The Http Session Binding Event Class,<br>**Handling HTTP Requests and Responses, Handling**: HTTP GET Requests,<br>Handling HTTP POST Requests, Using Cookies, Session Tracking. | **11 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books:**

1. Herbert Schildt , The Complete Reference, JAVA 7th/9th Edition, Tata McGraw Hill, 2013.
2. Java 6 Programming Black Book, Dreamtech Press. 2012

**Reference Books:**

1. Java Fundamentals: A comprehensive Introduction by Herbert Schildt, Dale Skrien. Tata McGraw Hill Edition 2013.
2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education,2004.
3. Uttam K Roy, Advanced JAVA programming, Oxford University press, 2015.

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS43 | CO1 | Apply the concepts of programming and implement programs using Java constructs. |
| | CO2 | Create classes and demonstrate object oriented programming concepts |
| | CO3 | Demonstrate inheritance , overloading and run-time errors using exception handling mechanism. |
| | CO4 | Develop GUI application program using Applet , event handling and database connectivity. |
| | CO5 | Design and develop web application using servelet programming. |

| **Course Title: ANALYSIS AND DESIGN OF ALGORITHM** | | |
|---|---|---|
| Subject Code : **19CS44** | Credit : 4 | CIE: 50 |
| Number of Lecture Hours/Week | **04 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | **52** | SEE Hours: 03 |
| **Pre-requisites:** Data structures using C. | | |
| **Course objectives:**<br>• Analyze the asymptotic performance of the algorithms in time and space domain.<br>• Introduce various algorithm design techniques. | | |

| **Modules** | **Teaching Hours** |
|---|---|
| **Module-I**<br>Algorithm, Fundamentals of Algorithmic Problem Solving, Important problem Types, Fundamental of Data Structures, Fundamentals of the Analysis of Algorithm Efficiency; Analysis Framework, Asymptotic Notations , Basic Efficiency Classes, Non-recursive and Recursive Algorithms, Examples-Fibonacci Numbers. | **10 Hrs** |
| **Module-II**<br>**Brute Force:** Introduction, Selection sort, Bubble Sort, Sequential search Brute-Force String Matching Exhaustive Search<br>**Divide & Conquer :** Introduction, Merge Sort, Quick Sort, Binary Search, Binary tree traversals & related properties, Multiplication of large integers & Stressen's Matrix Multiplication Insertion Sort. | **11 Hrs** |
| **Module-III**<br>**Decrease & Conquer** : Introduction, Depth First search, Breadth First Search, Topological Sorting, Algorithms for Generating Combinatorial objects.<br>**Transform & Conquer :** Introduction , Presorting, Balanced Search Trees, 2-3 Trees, Heaps and Heap Sort, Problem Reduction, **Space & Time Tradeoffs :** Sorting by Counting, Input Enhancement in String matching , Hashing. | **11 Hrs** |
| **Module-IV**<br>**Dynamic Programming:** Introduction, Computing a Binomial Coefficient, Warshall's Algorithm, Floyd's Algorithm, The Knapsack Problem and Memory Functions.<br>**Greedy Techniques**: Introduction, Minimum Spanning Tree, Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffmancodes. | **10 Hrs** |
| **Module-V**<br>**Limitations of Algorithms Power:** Introduction, Lower- Bound Arguments, Decision Trees, P, NP, and NP – Complete Problems.<br>**Backtracking:** Introduction, n-Queen‟s problem, Hamiltonian circuit problem, Subset problem, General backtracking algorithm, **Branch- and-Bound:** The assignment problem , Knapsack problem, Travelling sales man problem. | **10 Hrs** |

**Question paper pattern:**

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

**Text books:**

1. Anany Levitin, " Introduction to the Design & Analysis of Algorithm ", 2nd Edition, Pearson Edition, 2007.

**Reference Books:**

1. Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, "Introduction Algorithm", 2nd Edition, PHI,2006.
2. Horowitz E, Sahni S., Rajasekaran S., "Computer Algorithms", Galgotia Publications,2001.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS44 | **CO1** | Explain fundamental ideas used for designing and analyzing algorithms. |
| | **CO2** | Demonstrate Brute Force, Divide-and-Conquer techniques and analyze the performance of algorithms. |
| | **CO3** | Demonstrate design of Decrease & Conquer and Transform & Conquer algorithms and their efficiencies. |
| | **CO4** | Apply Dynamic Programming and Greedy Techniques to solve various graph problems efficiently. |
| | **CO5** | Describe Limitations of algorithms power and illustrate Back tracking, Branch-and-Bound algorithms to solve recursive and computational problems. |

| Course Title: **MICROPROCESSOR AND MICROCONTROLLER** | | |
|---|---|---|
| Subject Code : **19CS45** | Credit : 4 | CIE: 50 |
| Number of Lecture Hours/Week | **04 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |
| **Pre-requisites:** Logic Design, Basic Electronics | | |
| **Course objectives:**<br>• Explore the microprocessor architecture and its instruction set.<br>• Develop skills for programming in Assembly language.<br>• Interface Peripheral devices with 8086 Microprocessor and ARM Processor | | |

| **Modules** | **Teaching Hours** |
|---|---|
| **Module-I**<br>**Basic Structures of Computers, machine instructions & programs:** Basic Operational Concepts : Bus Structures, Performance, Processor clock, Basic Performance equation, Pipelining & Superscalar operation, Clock rate, Performance measurement, Multiprocessor & Multicomputer.<br>The x86 microprocessor: Inside the 8088/86, Introduction to assembly programming, Introduction to Program Segments, The Stack, Flag register, x86 Addressing Modes. Assembly language programming: Directives & a Sample Program, Assemble, Link & Run a program, More Sample programs, Control Transfer Instructions, Data Types and Data Definition, Full Segment Definition. | **12 Hrs** |
| **Module-II**<br>**x86:** Instructions sets description, Arithmetic and logic instructions and programs: Unsigned Addition and Subtraction, Unsigned Multiplication and Division, Logic Instructions, BCD and ASCII conversion, Rotate Instructions. INT 21H and INT 10H Programming : Bios INT 10H Programming , DOS Interrupt 21H. 8088/86 Interrupts, x86 PC and Interrupt Assignment. | **11 Hrs** |
| **Module-III**<br>**Signed Numbers and Strings:** Signed number Arithmetic Operations, String operations. Memory and Memory interfacing: Memory address decoding, data integrity in RAM and ROM, 16-bit memory interfacing. 8255 I/O programming: I/O addresses MAP of x86 PC‟s, programming and interfacing the 8255. | **10 Hrs** |
| **Module-IV**<br>**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 | **10 hrs** |

| **Module-V**<br>**Introduction to the ARM Instruction Set :** Data Processing Instructions , Branch Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions, Loading Constants, Simple programming exercises. | **09 Hrs** |
|---|---|

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text books:**
1. Carl Hamacher, Z.Vranesic & S.Zaky, Computer Organization, 5th Edition , Tata McGraw-Hill Publishing Company Ltd. New Delhi, 2011.
2. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Danny Causey, The x86 PC Assembly Language Design and Interfacing, 5th Edition, Pearson,2013.
3. Andrew N Sloss, Dominic Symes and Chris Wright, ARM system developers guide, Elsevier, Morgan Kaufman publishers,2008.

**Reference Books:**
1. Douglas V. Hall: Microprocessors and Interfacing, Revised 2nd Edition, TMH,2006.
2. K. Udaya Kumar & B.S. Umashankar : Advanced Microprocessors & IBM-PC Assembly Language Programming, TMH 2003.
3. Ayala : The 8086 Microprocessor: programming and interfacing - 1st edition, Cengage Learning
4. Joseph Yiu, The Definitive Guide to the ARM Cortex-M3, 2nd Edition , Newnes,2009
5. The Insider‟s Guide to the ARM7 based microcontrollers, Hitex Ltd.,1st edition,2005
6. Steve Furber, ARM System-on-Chip Architecture, Second Edition, Pearson,2015
7. Architecture, Programming and Interfacing of Low power Processors- ARM7, Cortex-M and MSP430, Lyla B Das Cengage Learning, 1stEdition

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS45** | **CO1** | Analyze the 8086 processor Structure, Assembly Language Programming and System programs used in Assembly language programming. Acquire knowledge on basic structure of computer and its performance |
| | **CO2** | Develop assembly language code to solve problems |
| | **CO3** | Design hardware interfacing of memory devices to x86 family |
| | **CO4** | Compare Microprocessor and Microcontroller, Explain interfacing through ARM processor, interrupt routines |
| | **CO5** | Demonstrate Instruction set and develop programs using ARM processor |

| **Course Title: ANALYSIS AND DESIGN OF ALGORITHM LAB** | | |
|---|---|---|
| Subject Code : **19CSL41** | Credits : 1 | CIE: 50 |
| Number of Lecture Hours/Week | **2 Hrs (Practical)** | SEE: 50 |
| | | SEE Hours: 03 |

**Prerequisite:** C Language : Functions and Recursion

**Corse Objectives :** To **e**nable the students for

- Learn different searching and sorting techniques.
- Gain knowledge of binary tree principles.
- Understand the different algorithms to solve the problems.

**PART – A**

**Using C / C++**

1. Write a C Program to implement Recursive Binary search and linear search and determine the time required to search an element.
2. Write a C Program to sort a given set of elements using Merge sort method and determine the time required to sort the elements.
3. Write a C Program to Sort a given set of elements using Selection sort and determine the time required to sort elements.
4. Write a C Program to Sort a given set of elements using Insertion sort and determine the time required to sort elements.
5. Write a C Program to Sort a given set of elements using the Heap sort method and determine the time required to sort the elements.
6. Write a C Program to Sort a given set of elements using Quick sort method and determine the time required sort the elements.
7. Write a C Program to Print all the nodes reachable from a given starting node in a digraph using BFS method.
8. Write a C Program to Check whether a given graph is connected or not using DFS method.

**PART – B**

1. Write a C Program to Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.
2. Write a C Program to Find Minimum Cost Spanning Tree of a given undirected graph using Prim‟s algorithm.

| | |
|---|---|
| 3. Write a C Program to From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra'salgorithm.<br><br>4. Write a C Program to implement 0/1 Knapsack problem using dynamic programming.<br><br>5. Write a C Program to 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{1,2,6}and{1,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.<br><br>6. Write a C Program to Implement Horspool algorithm for String Matching.<br><br>7. Write a C Program to Find the Binomial Co-efficient using Dynamic Programming.<br>8. Write a C Program to Implement Floyd‟s algorithm for the All-Pairs-Shortest- Paths problem.<br>9. Write a C Program to Compute the transitive closure of a given directed graph using Warshall‟s algorithm.<br><br>10. Write a C Program to Implement N Queen's problem using Back Tracking. | |

**Question paper pattern:**

**Note : For SEE, students will be asked to execute two programs, selecting one program from each part.**

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL41 | CO1 | Apply the knowledge of Divide-and-Conquer techniques for different searching and sorting problems using recursive method and find the time complexity of algorithms. |
| | CO2 | Demonstrate Decrease-and-Conquer techniques for solving thegraph problems. |
| | CO3 | Design and implement algorithms for solving the graph problems by using Greedy techniques. |
| | CO4 | Demonstrate the concepts of Dynamic Programming techniques by calculating the Binomial Co-efficient. |
| | CO5 | Illustrate the Back Tracking algorithms for subset and N- Queen‟s problems. |

| **Course Title: MICROPROCESSOR AND MICROCONTROLLER LAB** | | |
|---|---|---|
| Subject Code : **19CSL42** | Credits : 1 | CIE: 50 |
| Number of Lecture Hours/Week | **2 Hrs (Practical)** | SEE: 50 |
| | | SEE Hours: 03 |
| **Prerequisite:** C Programming | | |
| **Course Objectives :** Explore the microprocessor architecture and instruction set | | |

| **List of Programs** | |
|---|---|
| **SOFTWARE PROGRAMS: PART A**<br><br>1. Design and develop an assembly language program to search a key element "X" in a list of „n" 16-bit numbers. Adopt Linear search algorithm in your program for searching.<br><br>2. Design and develop an assembly program to sort a given set of „n" 16-bit numbers in ascending order. Adopt Bubble Sort algorithm to sort given elements.<br><br>3. Develop an assembly language program to reverse a given string and verify whether it is a palindrome or not. Display the appropriate message.<br><br>4. Develop an assembly language program to compute nCr using recursive procedure. Assume that „n" and „r" are non-negative integers.<br><br>5. Design and develop an assembly language program to read the current time and Date from the system and display it in the standard format on the screen.<br><br>**HARDWARE PROGRAMS: PART B**<br><br>6. Design and develop an assembly program to interface 4*4 matrix keyboard. using ARM TTDMI/LPC2148.<br>7. Design and develop an assembly program to implement the buzzer using ARMTTDMI/LPC2148 | |

| | |
|---|---|
| 8. Design and develop an assembly program to drive a Stepper Motor interface and rotate the motor in specified direction (clockwise or counter- clockwise) by N steps (Direction and N are specified by the examiner). Introduce suitable delay between successive steps. (Any arbitrary value for the delay may be assumed by the student) using ARM TTDMI/LPC2148.<br><br>9. Design and develop an assembly language program to<br>a. Generate the Sine Wave using DAC interface (The output of the DAC is to be displayed on the CRO).<br>b. Generate a Half Rectified Sine waveform using the DAC interface. ) using ARM TTDMI/LPC2148.<br><br>10. To interface LCD with ARM processor0 ARM7TDMI/LPC2148. Write and execute programs in C language for displaying text messages and numbers on LCD.<br>**Study Experiments:**<br>1. Interfacing of temperature sensor with ARM freedom board (or any other ARM microprocessor board) and display temperature on LCD<br>2. To design ARM cortex based automatic number plate recognition system<br>3. To design ARM based power saving system | |

**Question paper pattern:**

Conduction of Practical Examination:

- All laboratory experiments are to be included for practical examination.
- The board layout and the circuit diagram of the interface are to be provided to the student during the examination.
- Software Required: Open source ARM Development platform, KEIL IDE and Proteus for simulation

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL42 | CO1 | Develop ALP for fixed and Floating Point and Arithmetic operations using 8086 microprocessor. |
| | CO2 | Design and develop assembly programs using 8086 DOS functions in assembly language |
| | CO3 | illustrate how the different peripherals (8255, 8253 etc.) are interfaced with Microprocessor. |
| | CO4 | Design circuits for various applications using ARM microcontrollers |
| | CO5 | Construct different waveforms using interfacing 8086 microprocessor |

| **Course Title: Java Programming Laboratory** | | |
|---|---|---|
| Subject Code : **19CSL43** | Credit : 1 | CIE: 50 |
| Number of Lecture Hours/Week | 2 Hrs (Practical) | SEE: 50 |
| Total Number of Lecture Hours | | SEE Hours: 03 |
| **Prerequisites: Concepts of Object Oriented Programming, Programming** | | |
| **Course Objectives:** Learn to code and execute Java programs to solve problems, design of GUI for Java applications, Servlets for web applications, and database connectivity. | | |
| **MODULES** | | **Teaching Hours** |

**Preliminary practice programs**:

i) Understand and acquaint with Eclipse IDE environment. Write and execute a Java program to store and access student information.

ii) Write and execute a Java program to calculate sum of series of naturalnumbers

iii) Write and execute a Java program to demonstrate the scope of variables.

iv) Write and execute a Java program to find the biggest name in the array ofstrings.

v) Write and execute a Java program to demonstrate data type casting.

**Regular Laboratory exercises (for SEE):**

**(Every program should be a separate project and a package in EclipseIDE)**

1. Write and execute a JAVA program to demonstrate use of any five stringfunctions. Use both parameterized and non-parameterized constructors forpassing string inputs.

2. Write and execute a JAVA Program to demonstrate Inheritance.

3. Write and execute a JAVA Program to demonstrate exception handling (bothbuilt- in and user-defined exceptions).

4. Write and execute a JAVA Program to implement inheritance (single leveland multilevel).

**5.** Write and execute a JAVA program to demonstrate polymorphism throughmethod overloading.

**6.** Write and execute a JAVA program to demonstrate method overriding.

| | |
|---|---|
| 7. Write a JAVA applet program and required HTML file to create banner applet.<br><br>8. Write a JAVA applet program to create a basic Applet having buttons, text area GUI controls to add & subtract two nos. Use appropriate event listeners.<br><br>9. Write a Java program to store, delete and update data in a database with the support of JDBC-ODBC connectivity.<br><br>10. Write a Java program with Servlets to store only valid data in a database with the support of JDBC-ODBC connectivity.<br><br>11. Write a JAVA Servlet program to create login page for authentication purpose.<br><br>12. Write a JAVA Servlet program for adding cookies to HTML page for counting number of visits to a web page. | |

**Question paper pattern:**
**Note : For SEE, students will be asked to execute two programs, selecting one program from each part.**

**REFERENCES:**
www.tutorialpoint.com , www.w3schools.com

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL43 | CO1 | Implement Java programs with basic concepts of Object oriented programming. |
| | CO2 | Demonstrate Run-time and user-defined exceptions, Constructors. |
| | CO3 | Develop code for Inheritance, method overriding and overloading |
| | CO4 | Design interactive GUI Java programs using applets and database connectivity |
| | CO5 | Create Servlet for web applications |

# CURRICULUM
# FOR B.E.V SEMESTER

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING


## POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING
## (An Autonomous College under VTU)
## KALABURAGI

**About the Institution:** The Hyderabad Karnataka Education (HKE) society founded by Late Shri Mahadevappa Rampure, a great visionary and educationist. The HKE Society runs 46 educational institutions. Poojya DoddappaAppa College of Engineering, Gulbarga is the first institution established by the society in 1958. The college is celebrating its golden jubilee year, setting new standards in the field of education and achieving greater heights. The college was started with 50% central assistance and 50% state assistance, and a desire to impart quality technical education to this part of Karnataka State. The initial intake was 120 with degree offered in three branches of engineering viz, Civil, Mechanical and Electrical Engineering. Now, it houses 11 undergraduate courses, 10 post Graduate courses and 12 Research centers, established in Civil Engg., Electronics & Communication Engg, Industrial & Production Engg, Mechanical Engg, Electrical Engg., Ceramic Cement Tech., Information Science & Engg., Instrumentation Technology, Automobile Engg., Computer Sc. and Engg., Mathematics and Chemistry All the courses are affiliated to Visveswaraya Technological University, Belgaum. At present the total intake at UG level is 980 and PG level 193.

The college receives grant in aid funds from state government. A number of projects have been approved by MHRD /AICTE, Govt. of India for modernization of laboratories. KSCST, Govt. of Karnataka is providing financial assistance regularly for the student's projects.

The National Board of Accreditation, New Delhi, has accredited the College in the year 2005-08 for 09 UG Courses out of which 08 courses are accredited for three years and 01 course is accredited for five years. And second time accredited for Six Course in the year 2009- 2012

Our college is one among the 14 colleges selected under TEQIP, sponsored by World Bank.It has received a grant of Rs 10.454 Crores under this scheme for its development. The institution is selected for TEQIP phase II in year 2011 for four years. Institution is receiving a grant of Rs 12.50 Crores under TEQIP Phase -II scheme for its development and selected for TEQIP-III as mentoring Institute for BIET Jhansi(UP).

Recognizing the excellent facilities, faculty, progressive outlook, high academic standards and record performance, the VTU Belgaum reposed abundant confidence in the capabilities of the College and the College was conferred Autonomous Status from the academic year 2007-08, to update its own programme and curriculum, to devise and conduct examinations, and to evaluate student's performance based on a system of continuous assessment. The academic programmers are designed and updated by a Board of Studies at the department level and Academic Council at the college level. These statutory bodies are constituted as per the guidelines of the VTU Belgaum. A separate examination section headed by a Controller of Examinations conducts the examinations.

At present the college has acquired the Academic autonomous status for both PG and UG courses from the academic year 2007-08 and it is one among the six colleges in the state of Karnataka to have autonomous status for both UG and PG courses.

One of the unique features of our college is, it is the first college in Karnataka State to start the Electronics and Communication Engineering branch way back in the year 1967, to join NIT Surathkal and IISc, Bangalore. Also, it is the only college in the state and one among the three colleges across the country, offering a course in Ceramic and Cement Technology. This is the outcome of understanding by faculty and management about the basic need of this region, keeping in view of the available raw material and existing Cement Industries.

Bharatiya Vidya Bhavan National Award for an Engineering College having Best Overall Performance for the year 2017 by ISTE (Indian Society for Technical Education). In the year 2000, the college was awarded as Best College of the year by KSCST, Bangalore in the state level students projects exhibition.

The college campus is spread over 71 acres of land on either side of Mumbai-Chennai railway track and has a sprawling complex with gardens and greenery all around.

**About the department:** The Computer Science and Engineering department was started in the year 1984 with an intake of 40 students for UG. The department has seen phenomenal growth and now the department has increased UG intake to 120 students and offering two Post Graduation programmes : PG (Computer Science and Engineering with an intake of 25 students) and PG(Computer Network and Engineering with an intake of 18 students). The department is offering research program under its recognized research center. Computer Science and Design course was started from 2021 with an intake of 60 students. The department is having state-of-the-art computing facilities with high speed internet facilities and laboratories. The department library provides useful resources like books and journals. The department has well qualified and experienced teaching faculty. The department has been conducting several faculty development programs and student training programs.

## Vision of the Institution

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

## Mission of the Institution

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

## Vision of the Department

- To become a premier department in Computer education, research and to prepare highly competent IT professionals to serve industry and society at local and global levels.

## Mission of the Department

- To impart high quality professional education to become a leader in Computer Science and Engineering.
- To achieve excellence in Research for contributing to the development of the society.
- To inculcate professional and ethical behaviour to serve the industry.

**Program Educational Objectives (PEO)**:

| | |
|---|---|
| **PEO1:** | To prepare graduates with core competencies in mathematical and engineering fundamentals to solve and analyze computer science and engineering problems |
| **PEO2:** | To adapt to evolving technologies and tools for serving the society |
| **PEO3:** | To perform as team leader, effective communicator and socially responsible computer professional in multidisciplinary fields following ethical values |
| **PEO4:** | To encourage students to pursue higher studies, engage in research and to become entrepreneurs |

**Program Outcomes:**

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

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

03. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

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

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

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

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

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

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

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

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

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

**Program Specific Outcomes (PSOs):**

| | |
|---|---|
| **PSO1:** | Acquire competency in hardware and software working principles to analyze and solve computing problems. |
| **PSO2:** | Design quality software to develop scientific and business applications following Software Engineering practices. |
| **PSO3:** | Apply cutting edge technologies using modern tools to find novel solutions ethically to existing problems. |

## SCHEME OF TEACHING FOR V SEMESTER - 2022-2023

### B.E. (COMPUTER SCIENCE AND ENGINEERING)

| Code No. | Course | Teaching Hours/Week | | | Examination | | | |
|---|---|---|---|---|---|---|---|---|
| | | Lecture | Tutorial | Practical | CIE | SEE | Total | Credits |
| 19CS51 | Python Programming | 4 | 0 | 0 | 50 | 50 | 100 | **4** |
| 19CS52 | Finite Automata and Formal Language | 4 | 0 | 0 | 50 | 50 | 100 | **4** |
| 19CS53 | Database Management System | 4 | 0 | 0 | 50 | 50 | 100 | **4** |
| 19CS54 | Software Engineering and Testing | 3 | 0 | 0 | 50 | 50 | 100 | **3** |
| 19CS55 | Artificial Intelligence | 3 | 0 | 0 | 50 | 50 | 100 | **3** |
| 19HU01 | Recruitment Process training - I | 2 | 0 | 0 | 50 | 50 | 100 | **1** |
| 19CSL51 | Python Programming Lab | 0 | 0 | 2 | 50 | 50 | 100 | **1** |
| 19CSL52 | Database Management System Lab | 0 | 0 | 2 | 50 | 50 | 100 | **1** |
| 19CSL53 | Web Application Development Lab | 0 | 2 | 2 | 50 | 50 | 100 | **2** |
| | **Total** | **20** | **2** | **6** | **450** | **450** | **900** | **23** |

**AUTONOMOUS SYLLABUS FOR B.E V SEMESTER 2022-2023**

| **Course Title**: **PYTHON PROGRAMMING** | | |
|---|---|---|
| Subject Code : **19CS51** | **Credit : 4** | CIE: 50 |
| Number of Lecture Hours/Week | **4 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |
| **Prerequisites:** Knowledge about programming languages like C or C++ and Oops concepts**.** | | |
| **Course Objectives:**<br>• To know the basics of algorithmic problem solving<br>• To construct and write simple Python programs.<br>• To develop Python programs with conditionals and loops.<br>• To define Python functions and call them.<br>• To use Python data structures - lists, tuples, dictionaries.<br>• To do input/output with files in Python. | | |
| **MODULES** | | **Teaching Hours** |
| **Module I**<br>**DATA, EXPRESSIONS, STATEMENTS:**<br>Python interpreter and interactive mode; values and types: int, float, boolean, string, and list; variables, expressions, statements, tuple assignment, precedence of operators, comments; modules and functions, function definition and use, flow of execution, parameters and arguments; Illustrative programs: exchange the values of two variables, circulate the values of n variables, distance between two points. | | **10 Hrs** |
| **Module II**<br>**CONTROL FLOW, FUNCTIONS :**<br>Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional (if-elif-else); Iteration: state, while, for, break, continue, pass; Fruitful functions: return values, parameters, local and global scope, function composition, recursion; Strings: string slices, immutability, string functions and methods, string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum an array of numbers, linear search, binary search. | | **10 Hrs** |
| **Module III**<br>**LISTS, TUPLES, DICTIONARIES, OBJECT AND CLASSES:**<br>**Lists:** list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters; Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing - list comprehension. File handling and exception handlings. Objects and Classes, Classes and Functions, Classes and Methods, Inheritance. | | **11 Hrs** |
| **Module IV**<br>**NumPy:** Introduction to NumPy**:** Understanding Data Types in Python, The Basics of NumPy Arrays, Computation on NumPy Arrays: Universal Functions, Aggregations: Min, Max, and Everything In Between, Computation on Arrays: Broadcasting, Comparisons, Masks, and Boolean Logic, Fancy | | **11 Hrs** |

| | |
|---|---|
| Indexing, Sorting Arrays, Structured Data: NumPy's Structured Arrays.<br>**Visualization with Matplotlib:** Simple Line Plots, Simple Scatter Plots, Visualizing Errors, Density and Contour Plots, Histograms, Binnings and Density. | |
| **Module V**<br>**Pandas:** Data Manipulation with Pandas, Introducing Pandas Objects, Data Indexing and Selection, Operating on Data in Pandas, Handling Missing Data, Hierarchical Indexing, Combining Datasets: Concat and Append, Combining Datasets: Merge and Join, Aggregation and Grouping, Pivot Tables, Vectorized String Operations, Working with Time Series, High-Performance Pandas: eval() and query(). | **10 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**


1. N. Dasharanathan, K.Saraswati, R Rekha , S.R. Baseline Prabhu, Problem solving and python programming , Education Publication.
2. Jake Vanderplas, "Python Data Science Handbook", O"Reilly Publications, ISBN: 9781491912058, 2016. **(For Module IV and V).**
3. Bharti Motwani, "Data Analytics using Python", Wiley , 2020.


**REFERENCES:**


1. Charles Dierbach, "Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition, 2013.
2. John V Guttag, "Introduction to Computation and Programming Using Python"", Revised and expanded Edition, MIT Press , 2013
3. Kenneth A. Lambert, "Fundamentals of Python: First Programs", CENGAGE Learning, 2012.
4. Paul Gries, Jennifer Campbell and Jason Montojo, "Practical Programming: An Introduction to Computer Science using Python 3", Second edition, Pragmatic Programmers,LLC,2013.
5. Robert Sedgewick, Kevin Wayne, Robert Dondero, "Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.
6. Timothy A. Budd, "Exploring Python", Mc-Graw Hill Education (India) Private Ltd. 2015.


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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS51** | CO1 | Illustrate python list, tuples, dictionaries for representing compound data. |
| | CO2 | Illustrate a Python program using control statement and functions. |
| | CO3 | Represent compound data using Python data structure and demonstrate the use of OOPs concepts in python programming. |
| | CO4 | Applying NumPy for Data science applications. |
| | CO5 | Demonstrate the use of Pandas. |

| Course Title: **FINITE AUTOMATA AND FORMAL LANGUAGE** | | |
|---|---|---|
| Subject Code :**19CS52** | **Credit : 4** | CIE: 50 |
| Number of Lecture Hours/Week | **4 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |

**Pre-requisites:** Mathematical Foundations of Computer Science

**Course objectives:**

- To gain an understanding of automata theory principles
- Familiarize applications of automata theory in compiler construction and text processing.

| **Modules** | **Teaching Hours** |
|---|---|
| **Module-I**<br>**Introduction to finite automata:** Introduction to Finite Automata, The central concepts of Automata theory; Deterministic finite automata, Nondeterministic finite automata, An application of finite automata, Finite automata with Epsilon-transitions. | **11 Hrs** |
| **Module-II**<br>**Regular expressions, Regular languages and Properties:** Regular expressions; Finite Automata and Regular Expressions; Applications of Regular Expressions. **Regular languages and properties:** Regular languages; Proving languages not to be regular languages, Closure properties of regular languages. | **10 Hrs** |
| **Module-III**<br>**Properties of regular languages contd. , Context free grammars:** Decision properties of regular languages, Equivalence and minimization of automata. **Context-free grammars and languages:** Context –free grammars; Parse trees; Applications; Ambiguity in grammars and Languages. | **10 Hrs** |
| **Module-IV**<br>**Pushdown automata:** Definition of the Pushdown automata; The languages of a PDA; Equivalence of PDA‟s and CFG‟s; Deterministic Pushdown Automata. Properties of context-free languages: Normal forms for CFGs; The pumping lemma for CFGs; Closure properties of CFL. | **11 hrs** |
| **Module-V**<br>**Introduction to Turing machine:** Problems that Computers cannot solve; The turning machine; Programming techniques for Turning Machines; Extensionsto the basic Turning Machines; Turing Machine and Computers.<br>**Undecideability: A** Language that is not recursively enumerable; An Undecidable problem that is RE; Post‟s Correspondence problem; Other undecidable problems. | **10 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text books:**

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

**Reference Books:**

1. Raymond Greenlaw, H.JamesHoove, Morgan Kaufmann, Fundamentals of the Theory of Computation: Principles and Practice –, 1998.
2. John C Martin, Introduction to Languages and Automata Theory –3rd Edition, Tata McGraw-Hill, 2007.
3. Daniel I.A. Cohen, Introduction to Computer Theory –2nd Edition, John Wiley & Sons, 2004.
4. Thomas A. Sudkamp,An Introduction to the Theory of Computer Science, Languages and Machines –3rdEdition, Pearson Education, 2006.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS52 | CO1 | Design Deterministic and non Deterministic finite automata for agiven language and identify related applications in text processing. |
| | CO2 | Construct Regular expressions for given language and describe properties of regular language. |
| | CO3 | Develop Context Free Grammar and illustrate with its applications |
| | CO4 | Design PDA, discuss equivalence of CFG and PDA and explain properties of Context Free Languages. |
| | CO5 | Illustrate Turing machine concepts and its variants and the notion of undecidability. |

| **Course Title: DATABASE MANAGEMENT SYSTEM** | | |
|---|---|---|
| Subject Code : **19CS53** | Credit :4 | CIE: 50 |
| Number of Lecture Hours/Week | **4 hrs** | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |

**Prerequisites:** knowledge of C, C++ Programming Principles, Data Structures

**Course Objectives:**

- Learn and practice data modelling using entity relationship and developing database design
- Understand the use of SQL
- Understand the functional dependency and Normalization Techniques.
- Understand the online transaction processing and recovery methods.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Introduction:** An example, Characteristics of Database approach, Actors on the screen, Workers behind the scene, Advantages of using DBMS approach, A brief history of database applications, when not to use a DBMS. Data models, schemas and instances, Three-schema architecture and data independence, Database languages and interfaces. Entity-Relationship Model: Using High-Level Conceptual Data Models for Database Design, An Example Database Application, Entity Types, Entity Sets, Attributes and Keys, Relationship types, Relationship Sets, Roles and Structural Constraints, Weak Entity Types, Refining the ER Design, ER Diagrams, Naming Conventions and Design Issues, Relationship types of degree higher than two, Subclasses, Super Classes and Inheritance, Specialization and Generalization.<br>**Relational Model:** Relational Model Concepts, Relational Model Constraints and Relational Database Schemas. | **10 Hours** |
| **Module II**<br>**Introduction to SQL:** The SQL Language, The Role of SQL, SQL Features and Benefits, SQL and Networking (Centralized Architecture, File Server Architecture, Client/Server Architecture, Multi-Tier Architecture) The Relational Data Model: Tables, Primary Keys, Relationships, Foreign Keys, **SQL Basics:** Data Types, Constants, Expressions, Built-in Functions, Missing Data (NULL Values), Row Alias, Literals. **SQL Commands: DDL Statements:** Create, Alter, Drop, Truncate Tables, **DML Statements:** Insert, Update and Delete, **DCL Statements:** GRANT, REVOKE TCL Statements: COMMITT, ROLBACK, SAVEPOINT, Simple Queries: The **SELECT Statement**, The SELECT Clause, FROM Clause, WHERE Clause, **SQL Operators :**Arithmetic, Comparison, Logical operations on columns, **Other Operators:** BETWEEN AND, LIKE, IS NULL, IN Compound Search Conditions (AND, OR, and NOT), **Order by Clause:** Sorting Query Results, Combining Query Results (UNION) *,Unions and Sorting *, **Joins:** Simple Joins (Equi-Joins), Natural Join, Joins with Row Selection Criteria, Multiple Matching Columns, Table Aliases, **OUTER JOINS :**, Left and Right Outer Joins *, Inner Joins in SQL2 ***Aggregate Functions:** Column Functions: SUM() computes the total of a column. • AVG() computes the average value in a column. • MIN() finds the smallest value in a column. • MAX() finds the largest value in a column. • COUNT() counts the number of values in a column. • COUNT(*) counts rows of query results, Grouped Queries (GROUP BY Clause). | **12 hours** |

| | |
|---|---|
| **Module III**<br>**Database Design** - 1: Informal Design Guidelines for Relation Schemas, Functional Dependencies, And Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Code Normal Form. Database Design – 2: Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Multivalued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form, Inclusion Dependencies, Other Dependencies and Normal Forms. | **10 Hours** |
| **Module IV**<br>**Transaction Processing Concepts:** Introduction to Transaction Processing, 12 Transaction and System Concepts, Desirable Properties of Transactions, Characterizing Schedules Based on Recoverability, Characterizing Schedules Based on Serializability, Transaction Support in SQL. Concurrency Control Techniques: Two-Phase Locking Techniques for Concurrency Control, Concurrency Control Based on Timestamp Ordering, Multiversion Concurrency Control Techniques, Validation Concurrency Control Techniques, Granularity of Data items and Multiple Granularity Locking, Using Locks for Concurrency Control in Indexes. | **10 Hours** |
| **Module V**<br>**Database Recovery Techniques :** Recovery Concepts, Recovery Techniques Based on Deferred Update, Recovery Techniques Based on Immediate Update, Shadow Paging, The ARIES Recovery Algorithm, Recovery in Multi database Systems, Database Backup and Recovery from Catastrophic Failures. Database Security and Authorization: Introduction to Database Security Issues, Discretionary Access Control Based on Granting and Revoking Privileges, Mandatory Access Control and Role-Based Access Control for Multilevel Security, Introduction to Statistical Database Security, Introduction to Flow Control, Encryption and Public Key Infrastructures. | **10 Hours** |

**Question paper pattern:**

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

**Text books:**

1. Fundamentals of Database Systems - Elmasri and Navathe, 7th Edition,Addison- Wesley, 2016
2. SQL – The Complete Reference- James R Groff, Paul N.Weinberg and Andrew J.Oppel, 3rd Edition,Mc-Graw Hill, 2009. (Module-II)

**Reference Books:**

1. Data Base System Concepts- Silberschatz, Korth and Sudharshan, 5th Edition, Mc-Graw Hill, 2006.
2. Database Management Systems -Raghu Ramakrishnanand Johannes Gehrke – 3rd Edition. McGraw-Hill, 2003.
3. An Introduction to Database Systems - C.J. Date, A. Kannan, S. Swamynatham, 8th Edition, Pearson Education, 2006.

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS53 | CO1 | Understand the fundamentals and applications of data base management system. |
| | CO2 | Implement and Interact database with SQL statements. |
| | CO3 | Design data base by applying ER diagram, relational model, functional dependency and Normalization Techniques |
| | CO4 | Illustrate and understand the basic issues of transaction processing and concurrency control. |
| | CO5 | Demonstrate different recovery techniques and security issues. |

| **Course Title: SOFTWARE ENGINEERING AND TESTING** | | |
|---|---|---|
| Subject Code : **19CS54** | Credits : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| Prerequisites: Any programming language | | |

**Course objectives:**

- Acquire knowledge of software development life cycle
- Understand methodologies for designing the software
- Describe the development of efficient and cost effective software.
- Gain knowledge of Software Testing process.
- Perform various software testing and measurement.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module – I**<br>**Overview:** Introduction: FAQ's about software engineering, Professional and ethical responsibility.<br>**Software Processes:** Software Processes: Models, Process iteration, Process activities, The Rational Unified Process, Computer-Aided Software Engineering.<br>**Requirements:** Software Requirements: Functional and Non-functional requirements, User requirements, System requirements, Interface specification, and The software requirements document. | **08 Hrs** |
| **Module - II**<br>**Software Design:** Architectural Design: Architectural design decisions, System organization, Modular decomposition styles, Control styles. Object- Oriented design: Objects and Object Classes, An Object-Oriented design process, Design evolution, Introduction to UML Diagram, Case study **DEVELOPMENT:** Rapid Software Development: Agile methods, Extreme programming, Rapid application development, Software prototyping. | **09 Hrs** |
| **Module - III**<br>**Verification And Validation:** Verification and Validation: Planning, Software inspections, Automated static analysis, Verification and formal methods.<br>**Management**: Managing People: Selecting staff, Motivating people, Managing people, The People Capability Maturity Model.<br>**Software Cost Estimation:** Productivity, Estimation techniques. | **08 Hrs** |
| **Module – IV**<br>**A Perspective on Testing, Examples:** Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Levels of testing. Examples: Generalized pseudo code, The triangle problem, The Next Date function, The commission problem, The SATM (Simple Automatic Teller Machine) problem, The currency converter, Saturn windshield wiper. **Boundary Value Testing:** Boundary value analysis, Robustness testing, Worst-case testing, Special value testing, Examples, Random testing, Guidelines for Boundary value Testing. | **09 Hrs** |

| Module – V | |
|---|---|
| **Path Testing:** DD paths, Test coverage metrics, Basis path testing, guide lines and observations. Define/Use testing, Slice-based testing, Guidelines and observations.<br>**Levels of Testing:** Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing.<br>**Integration Testing:** A closer look at the SATM system, Decomposition-based Integration, call graph-based Integration. | **08 Hrs** |

**Question paper pattern:**

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

**Text book:**

1. **Software Engineering** – Ian Somerville, 10th Edition, Pearson Education,2016.
2. **Software Testing, A Craftsman's Approach** - Paul C. Jorgensen:, 4th Edition, Auerbach Publications,2013.

**Reference Books:**

1. **Software Engineering: A Practitioners Approach -** Roger S. Pressman, 7th Edition, McGraw-Hill,2007.
2. **Software Engineering Theory and Practice -** Shari Lawrence Pfleeger, Joanne M.Atlee, 3rd Edition, Pearson Education,2006.
3. **Software Engineering Principles and Practice -** Waman S Jawadekar, Tata McGraw Hill, 2004.
4. **Object Oriented System Development using UML,** Ali Bahrami , MaGrawHill, 1999

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS54 | CO1 | Describe software engineering process to account for quality issues and non-functional requirements. |
| | CO2 | Translate specification into a design, and then realize that design practically, using an appropriate software engineering methodology. |
| | CO3 | Develop, maintain and evaluate large-scale software systems, To produce efficient, reliable, robust and cost-effective software solutions |
| | CO4 | Discuss the fundamental principles of Software Testing with lifecycle and essential functional test methods. |
| | CO5 | Perform Basic test design and measurement techniques. |

| **Course Title: ARTIFICIAL INTELLIGENCE** | | |
|---|---|---|
| Subject Code : **19CS55** | **Credit :03** | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |

**Prerequisites: Mathematical Foundations of Computer Science**

**Course Objectives:**

- To Apply a given AI technique to a given concrete problem
- To Implement non-trivial AI techniques in a relatively large system
- To understand uncertainty and Problem solving techniques.
- To understand various symbolic knowledge representation to specify domains and reasoning tasks of a situated software agent.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Introduction to Artificial Intelligence:** The AI Problems, The Underlying assumption, AI Technique, The Level of the model, Criteria for success. **Problems, problem spaces, and search:** Defining, the problem as a state space search, Production systems, Problem characteristics, Production system characteristics, Issues in the design of search programs. | **09 Hrs** |
| **Module II**<br>**Heuristic search techniques:** Generate-and-test, Hill climbing, Best-first search, Problem reduction, Mean-ends analysis.<br>**Knowledge representation issues:** Representations and mappings, Approaches to knowledge representation, Issues in knowledge representation, the frame problem. | **08 Hrs** |
| **Module III**<br>**Using predicate logic:** Representing simple facts in logic, representing instance and ISA relationships, Computable functions and predicates, Resolution, Natural Deduction<br>**Representing Knowledge Using Rules:** Procedural versus Declarative knowledge, Logic programming, forward versus backward reasoning, matching, control knowledge. | **08 Hrs** |
| **Module IV**<br>**Symbolic Reasoning Under Uncertainty:** Introduction to non monotonic reasoning, Logic for non monotonic reasoning, Implementation Issues, Augmenting a problem-solver, Implementation: Depth-first search, Implementation: Breadth-first search.<br>**Statistical Reasoning:** Probability and bayes Theorem, Certainty factors and rule-based systems, Bayesian Networks, Dempster-Shafer Theory, Fuzzy logic. | **09 Hrs** |

| Module V | |
|---|---|
| **Text Analysis and Mining:** Introduction, Language Models, Text Classification, Information Retrieval, Information Extraction, Statistical Natural Language Processing, Cross- Lingual Natural Language Processing, Spell Checking, Speech Recognition.<br>**Expert System and Applications:** Expert System, Knowledge Representation, Expert System Shells, Knowledge Acquisition of an Expert system, Applications of expert Systems, Examples of Expert Systems, Problem Solving Examples. | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**

1. Elaine Rich and Kevin Knight, "Artificial Intelligence", Tata McGraw-Hill, 3rd Edition 2008
2. LavikaGoel, "Artificial Intelligence concepts and Applications", Wiley Emerging Technology Series 2021 by Wiley India Pvt.Ltd.

**REFERENCES:**

1. Nils J. Nilsson, "Artificial Intelligence: A new Synthesis", Harcourt Asia Pvt. Ltd.
2. George F. Luger, "Artificial Intelligence-Structures and Strategies for Complex Problem Solving", Pearson Education/ PHI.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS55** | **CO1** | Discuss artificial intelligence techniques, problem and heuristic search algorithm |
| | **CO2** | Apply knowledge representation techniques and predicate Logic rules to solve reasoning programs. |
| | **CO3** | Apply various symbolic reasoning under uncertainty in intelligent system development as well as understand the importance of maintaining intelligent systems. |
| | **CO4** | Discuss various learning methods using probabilistic models. |
| | **CO5** | Design and develop Natural Language Processing and Expert System applications. |

| Course Title: **PYTHON PROGRAMMING LAB** | | |
|---|---|---|
| Subject Code : **19CSL51** | Credit : 1 | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |

**Pre-requisites**: Knowledge of C and (or) C++ programming language, Concepts of Object oriented programming.

**Course Objectives:**

- Write, test, and debug simple Python programs.
- Implement Python programs with conditionals and loops.
- Develop Python programs step-wise by defining functions and calling them.
- Use Python lists, tuples, dictionaries for representing compound data.
- Read and write data from/to files in Python

**I. Practice programs**

1. Write a python program to add two numbers.
2. Check a number is positive/negative.
3. Find largest number among three numbers.
4. Store numbers in a array and find average of numbers.

**II Exercise Programs (for Lab IA and SEE lab exams)**

**A. Programs on Python built-in data structures**

1. Create a list and perform the following operations:
   Insert, remove, append, length, pop, show elements, remove all elements.
2. Create a dictionary and apply the following methods:
   Print all the dictionary items, Get the value of the specified key , update values, get values , get keys.
3. Create a tuple and perform the following methods:
   Find the length of tuple, Find the item, Access items, Find the number of times a specified value occurs in tuple.

**B. Programs using Functions:**

4. Write a program to check the given string is palindrome or not.
5. Write a program to find the Factorial of a given number.
6. Write a program to check equality of two lists by passing List as argument to function defined.
7. Write a program to double a given number and add two numbers using lambda().
8. Write a program for map() function to double all the items in the list.
9. Write a program to find sum of the numbers for the elements of the list by using reduce()

**C) Exception handling**

10. Demonstrate a python code to print try, except and finally block statements.

**D) Programs on File handling**

11. Write a python program to open and write, “hello world” into a text file and then append the same file with the text, “hi python programming” for the existing file.

**E) Working with OS, Calendar Module**

12. Write a python program to get python version and check the access permissions of file.
13. Write a python program to display a particular month of a year using calendar module and also all the months of the year.

**F) Packages**

14. Write a python program to create a package and use it in another program.

**G) Object oriented Python Programming**

15. Write a program to find sum of all the elements of a number list using class and methods
16. Write a program to demonstrate inheritance.
17. Write a program to demonstrate method overloading in python.

**H) Working with Numpy**

18. Write a program using a numpy module to create an array and check the following:
    1. Type of array
    2. Axes of array
    3. Shape of array
    4. Type of elements in array
19. Write a program using a numpy module to create an array and check the following:
    1. List with type float
    2. 3*4 array with all zeros
20. Write a program using a numpy module to :
    1. Reshape 3X4 array to 2X2X3 array
    2. Create Sequence of integers from 0 to 30 with steps of 5
    3. Flatten array
    4. Create Constant value array of complex type

**I) Working with Pandas**

21. Write a python program to concatenate the data frames with two different objects
22. Write a python code to read a csv file using pandas module and print the first and last five lines of a file.

**J) Working with Matplotlib**

23. Write a python program to demonstrate data visualization through plotting facilities available in matplotlib library on some sample data.

**Question paper pattern: For SEE , two programs from the Exercise programs list will be asked.**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL51 | **CO1** | Illustrate Python lists, tuples, dictionaries for representing compound data. |
| | **CO2** | Develop modular python programs by defining functions, exception and file handling. |
| | **CO3** | Demonstrate programming with system modules. |
| | **CO4** | Implement programs with object oriented concepts. |
| | **CO5** | Develop program to utilize Numpy and Pandas libraries for data analysis and visualize data with matplotlib library. |

| Course Title: **DATABASE MANAGEMENT SYSTEM LAB** | | |
|---|---|---|
| Subject Code : **19CSL52** | **Credit : 1** | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |

**Pre-requisites**: Knowledge of C, C++ Programming Principles, Data Structures

**Course Objectives:**

The student should be made to:

- Learn to create and use a database
- Be familiarized with a query language
- Have hands on experience on DDL Commands
- Have a good understanding of DML Commands and DCL commands
- Familiarize advanced SQL queries.
- Be Exposed to different applications.

**LIST OF EXPERIMENTS**:

**Part-A**

1. Draw E-R diagram and convert entities and relationships to relation table for a given scenario.
2. Perform the following: Viewing all databases, Creating a Database, Viewing all Tables in a Database.
3. Creating Tables (With and Without Constraints), Inserting/Updating/Deleting Records in a Table, Saving (Commit) and Undoing (rollback).
4. Perform the following: Altering a Table, Dropping/Truncating/Renaming Tables, Backing up / Restoring a Database.
5. For a given set of relation schemes, create tables and perform the following Simple Queries, Simple Queries with Aggregate functions, Queries with Aggregate functions (group by and having clause), Queries involving- Date Functions, String Functions , Math Functions Subqueries- With IN clause, With EXISTS clause
6. For a given set of relation tables perform the following: a. Creating Views (with and without check option), Dropping views, Selecting from a view
7. Write a Pl/SQL program using FOR loop to insert ten rows into a database table.
8. Illustrate how you can embed PL/SQL in a high-level host language such as C/Java and demonstrates how a banking debit transaction might be done.
9. Given an integer i, write a PL/SQL procedure to insert the tuple (i, 'xxx') into a given relation.
10. Write a PL/SQL block that handles all types of exceptions.

**Part-B**

1. Consider the following relations:

   Student (snum: integer, sname: string, major: string, level: string, age: integer)Class (name: string, meets at: string, room: string, d: integer)
   Enrolled (snum: integer, cname: string)
   Faculty (fid: integer, fname: string, deptid: integer)

   Write the following queries in SQL. No duplicates should be printed in any of the answers.

i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by Prof. Harshith
ii. Find the names of all classes that either meet in room R128 or have five or more Students
iii. Find the names of all students who are enrolled in two classes that meet at the same time.
iv. Find the names of faculty members who teach in every room in which some class is taught.
v. Find the names of faculty members for whom the combined enrollment of the courses that

they teach is less than five

2. The following relations keep track of airline flight information:
Flights (no: integer, from: string, to: string, distance: integer, Departs: time, arrives: time, price:real)
Aircraft (aid: integer, aname: string, cruising range: integer)Certified (eid: integer, aid: integer)
Employees (eid: integer, ename: string, salary: integer)
Note that the Employees relation describes pilots and other kinds of employees as well; Every pilotis certified for some aircraft, and only pilots are certified to fly.
**Write each of the following queries in SQL.**
   i. Find the names of aircraft such that all pilots certified to operate them have salaries more thanRs.80, 000.
   ii. For each pilot who is certified for more than three aircrafts, find the eid and the maximumcruising range of the aircraft for which she or he is certified.
   iii. Find the names of pilots whose salary is less than the price of the cheapest route from Bengaluru to Frankfurt.
   iv. For all aircraft with cruising range over 1000 Kms, .find the name of the aircraft and the averagesalary of all pilots certified for this aircraft.
   v. Find the names of pilots certified for some Boeing aircraft.
   vi. Find the aids of all aircraft that can be used on routes from Bengaluru to New Delhi.

3. Consider the following database of student enrollment in courses & books adopted for each course.
STUDENT (regno: string, name: string, major: string, bdate:date)COURSE (course #:int, cname:string, dept:string)
ENROLL ( regno:string, course#:int, sem:int, marks:int)
BOOK _ ADOPTION (course# :int,sem:int, book-ISBN:int)
TEXT (book-ISBN:int, book-title:string, publisher:string, author:string)
   i. Create the above tables by properly specifying the primary keys and the foreign keys.
   ii. Enter at least five tuples for each relation.
   iii. Demonstrate how you add a new text book to the database and make this book be adopted bysome department.
   iv. Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical orderfor courses offered by the „CS‟ department that use more than two books.
   v. List any department that has all its adopted books published by a specific publisher.
   vi. Generate suitable reports.
   vii. Create suitable front end for querying and displaying the results.

4. The following tables are maintained by a book dealer.
AUTHOR (author-id:int, name:string, city:string, country:string)
PUBLISHER (publisher-id:int, name:string, city:string, country:string)
CATALOG (book-id:int, title:string, author-id:int, publisher-id:int, category-id:int, year:int, price:int)
CATEGORY (category-id:int, description:string)

ORDER-DETAILS (order-no:int, book-id:int, quantity:int)
i. Create the above tables by properly specifying the primary keys and the foreign keys.
ii. Enter at least five tuples for each relation.
iii. Give the details of the authors who have 2 or more books in the catalog and the price of the
books is greater than the average price of the books in the catalog and the year of publication is
after 2000.
iv. Find the author of the book which has maximum sales.
v. Demonstrate how you increase the price of books published by a specific publisher by 10%.
vi. Generate suitable reports.
vii. Create suitable front end for querying and displaying the results.

5. Consider the following database for a banking enterprise
BRANCH(branch-name:string, branch- city:string, assets:real)
ACCOUNT(accno:int, branch-name:string, balance:real)
DEPOSITOR(customer-name:string, accno:int)
CUSTOMER(customer-name:string, customer-street:string, customer-city:string)
LOAN(loan-number:int, branch-name:string, amount:real)
BORROWER(customer-name:string, loan-number:int)
   i. Create the above tables by properly specifying the primary keys and the foreign keys.
   ii. Enter at least five tuples for each relation
   iii. Find all the customers who have at least two accounts at the Main branch.
   iv. Find all the customers who have an account at all the branches located in a specific city.
   v. Demonstrate how you delete all account tuples at every branch located in a specific city.
   vi. Generate suitable reports.
   vii. Create suitable front end for querying and displaying the results.

6. Consider the following schema for a Library Database:
BOOK(Book_id, Title, Publisher_Name, Pub_Year)
BOOK_AUTHORS(Book_id, Author_Name)
PUBLISHER(Name, Address, Phone)
BOOK_COPIES(Book_id, Programme_id, No-of_Copies)
BOOK_LENDING(Book_id, Programme_id, Card_No, Date_Out, Due_Date)
LIBRARY_PROGRAMME(Programme_id, Programme_Name, Address)
Write SQL queries to :
   i. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each Programme, etc.
   ii. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017.
   iii. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.

iv. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.

v. Create a view of all books and its number of copies that are currently available in the Library

7. Consider the following schema for Order Database:

SALESMAN(Salesman_id, Name, City, Commission)
CUSTOMER(Customer_id, Cust_Name, City, Grade, Salesman_id)
ORDERS(Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id)
Write SQL queries to

i. Count the customers with grades above Bangalore's average.

ii. Find the name and numbers of all salesman who had more than one customer.

iii. List all the salesman and indicate those who have and do not have customers in their cities (Use UNION operation.)

iv. Create a view that finds the salesman who has the customer with the highest order of a day.

v. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.

8. Consider the schema for College Database:

STUDENT(USN, SName, Address, Phone, Gender)SEMSEC(SSID, Sem, Sec)
CLASS(USN, SSID)
COURSE(Subcode, Title, Sem, Credits)
IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)
Write SQL queries to :

i. List all the student details studying in fourth semester 'C' section.

ii. Compute the total number of male and female students in each semester and in each section.

iii. Create a view of Test1 marks of student USN '1BI15CS101' in all Courses.

iv. Calculate the FinalIA (average of best two test marks) and update the corresponding table for allstudents.

v. Categorize students based on the following criterion:
If FinalIA = 17 to 20 then CAT =
Outstanding'If FinalIA = 12 to 16
then CAT = 'Average'
If FinalIA< 12 then CAT = 'Weak'

Give these details only for 8th semester A, B, and C section students

9. Consider the schema for Company Database:

EMPLOYEE(SSN, Name, Address, Sex, Salary, SuperSSN, DNo)
DEPARTMENT(DNo, DName, MgrSSN, MgrStartDate)
DLOCATION(DNo,DLoc)
PROJECT(PNo, PName, PLocation, DNo)
WORKS_ON(SSN, PNo, Hours)
Write SQL queries to

i. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.

ii. Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.

iii. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department.

iv. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator).

v. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.

10. Consider the schema for Movie Database:

ACTOR (Act_id, Act_Name, Act_Gender)
DIRECTOR (Dir_id, Dir_Name, Dir_Phone)
MOVIES (Mov_id, Mov_Title, Mov_Year,
Mov_Lang, Dir_id)
MOVIE_CAST (Act_id, Mov_id, Role)
RATING (Mov_id, Rev_Stars)

Write SQL queries to

i. List the titles of all movies directed by 'Hitchcock'.

ii. Find the movie names where one or more actors acted in two or more movies.

iii. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).

iv. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by Movie title.

v. Update rating of all movies directed by 'Steven Spielberg' to 5.

**Mini project (**Application Development using :Front end: VB/VC ++/JAVA or EquivalentBack end: Oracle / SQL / MySQL/ PostGress / DB2 or Equivalent).

a. Inventory Control System.
b. Core Banking system
c. Hospital Management System.
d. Railway Reservation System.
e. Personal Information System.
f. Web Based User Identification System.
g. Timetable Management System.
h. Hotel Management System.
i. Library management
j. Electricity bill.
k. Hostel management.
l. Air reservation
m. Company management system.
n. Student information system.
o. University database system.

**Guidelines for implementation of mini project**

1. Draw ER Diagram.
2. Convert ER diagram to table/schema.
3. Apply normalization.
4. Design and implementation
5. Generate report.

**Note: Mini Projects are to be considered only for CIE**
**Question paper pattern: For SEE , Students will be given programs from part B and Mini Project will be Evaluated**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL52 | CO1 | Design and implement a database schema for a given problem domain, Populate and query a database. |
| | CO2 | Design database using PL/SQL, Triggers, Exception Handling |
| | CO3 | Create and maintain tables using SQL. |
| | CO4 | Design database with constraints |
| | CO5 | Design and implement database for real world problem |

| **Course Title: WEB APPLICATION DEVELOPMENT LAB** | | |
|---|---|---|
| SubjectCode:**19CSL53** | Credit**: 2** | CIE:50 |
| Number of Tutorials Hours/Week | **2Hrs** | SEE:50 |
| Number of Practical Hours/Week | **2Hrs** | SEEHours:03 |
| **Prerequisites**: Java Object oriented concepts, Java Basics | | |

**Course Objectives:**

- Provide the principles and programming skills for development of Web applications.
- Enables students to develop skills for client/server programming and database applications Management.

**EXPERIMENTS**

1. Create an HTML documents to study various HTML tags, style sheets and the tag, Borders, padding, color, and the tag.

2. Develop a JavaScript embedded HTML file for.

   a) Generating Sum of n numbers. Use alert window to display the result

   b) Determine the roots of Quadratic Equation. Use document. Write to produce output.

3. Learn various array and object operations and perform the following operations:

   a) Create an empty array with name ‘todoList’

   b) Use ‘push’ operation on the ‘todoList’ array to add few objects each having ‘id’ as key and string as value (for ex {id:”a”},{id:”b”})

   c) Use ‘pop’ operation to remove the last element from the ‘todoList’ array.

   d) Use ‘filter’ operation to return a new array of objects with no object having id as “a”

4. Create a modal window using absolute positioning in CSS and use JavaScript for opening and closing the modal.

5. Learn basic flex commands and design a price card using flexbox for positioning of elements.

6. Design a website which dynamically adds and removes contents (To-Do list) using flexbox.

7. Analyze the working of CSS grid layout and create a website using grid layout.

8. Develop a weather website using REST API in JavaScript and use CSS Grid for positioning.

9. Install, configure, compare and discuss features of any open-source webserver, my SQL, PHP.

10) Write a PHP program to store current data-time in a COOKIE and display the Last visited on "date-time on the web page upon reopening the same page.

11. Using PHP and MySQL, develop a program to accept book information viz., Accession Number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.

12. On any HTML page, include a link for Login. Write a login page having login/password fields. Write JavaScript code to validate the login-id and password for the following: both are properly formed and at least 6 bytes long; the password contains at least one special case, one capital and one numeric character; convert the password into its MD5 hash use table created in experiment

13.Open ended experiment:
Using bootstrap tool develop an e commerce website.

**Question paper pattern: For SEE similar question related to the above programs will be asked.**

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

| Course Code | CO# | Course Outcome(CO) |
|---|---|---|
| 19CSL53 | CO1 | Design of Static web programming using HTML. |
| | CO2 | Create web pages using HTML, Cascading Style Sheets, JavaScript. |
| | CO3 | Design and implement dynamic Web pages with server side Information using Perl. |
| | CO4 | Design PHP programs for client server interaction. |
| | CO5 | Develop database applications using MySQL database with PHP. |

# CURRICULUM
# FOR B.E. VI SEMESTER

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING


## POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING
## (An autonomous college under VTU)KALABURAGI

**About the institution:** The Hyderabad Karnataka Education (HKE) society founded by Late Shri Mahadevappa Rampure, a great visionary and educationist. The HKE Society runs 46 educational institutions. Poojya DoddappaAppa College of Engineering, Gulbarga is the first institution established by the society in 1958. The college is celebrating its golden jubilee year, setting new standards in the field of education and achieving greater **heights. The** college was started with 50% central assistance and 50% state assistance, and a desire to impart quality technical education to this part of Karnataka State. The initial intake was 120 with degree offered in three branches of engineering viz, Civil, Mechanical and Electrical Engineering. Now, it houses 11 undergraduate courses, 10 post Graduate courses and 12 Research centers, established in Civil Engg., Electronics & Communication Engg, Industrial & Production Engg, Mechanical Engg, Electrical Engg., Ceramic Cement Tech., Information Science & Engg., Instrumentation Technology, Automobile Engg., Computer Sc. and Engg., Mathematics and Chemistry All the courses are affiliated to Visveswaraya Technological University, Belgaum. At present the total intake at UG level is 980 and PG level 193.

The college receives grant in aid funds from state government. A number of projects have been approved by MHRD /AICTE, Govt. of India for modernization of laboratories. KSCST, Govt. of Karnataka is providing financial assistance regularly for the student's projects.

The National Board of Accreditation, New Delhi, has accredited the College in the year 2005-08 for 09 UG Courses out of which 08 courses are accredited for three years and 01 course is accredited forfive years. And second time accredited for Six Course in the year 2009-2012

Our college is one among the 14 colleges selected under TEQIP, sponsored by World Bank. It has received a grant of Rs 10.454 Crores under this scheme for its development. The institution is selected for TEQIP phase II in year 2011 for four years. Institution is receivinga grant of Rs 12.50 Crores under TEQIP Phase -II scheme for its development and selected for TEQIP-III as mentoring Institute for BIET Jhansi(UP).

Recognizing the excellent facilities, faculty, progressive outlook, high academic standards and record performance, the VTU Belgaum reposed abundant confidence in the capabilities of the College and the College was conferred Autonomous Status from the academic year 2007-08, to update its own programme and curriculum, to devise and conduct examinations, and to evaluate student's performance based on a system of continuous assessment. The academic programmers are designed and updated by a Board of Studies at the department level and Academic Council at the college level. These statutory bodies are constituted as per the guidelines of the VTU Belgaum. A separate examination section headed by aController of Examinations conducts the examinations.

At present the college has acquired the Academic autonomous status for both PG and UG courses from the academic year 2007-08 and it is one among the six colleges in the state of Karnataka to have autonomous status for both UG and PG courses.

One of the unique features of our college is, it is the first college in Karnataka State to start the Electronics and Communication Engineering branch way back in the year 1967, to join NIT Surathkal and IISc, Bangalore. Also, it is the only college in the state and one among the three colleges across the country, offering a course in Ceramic and Cement Technology. This is the outcome of understanding by faculty and management about the basic need of this region, keeping inview of the available raw material and existing Cement Industries.

Bharatiya Vidya Bhavan National Award for an Engineering College having Best Overall Performance for the year 2017 by ISTE (Indian Society for Technical Education). In the year 2000, the college was awarded as Best College of the year by KSCST, Bangalore in the state level students projects exhibition.

The college campus is spread over 71 acres of land on either side of Mumbai-Chennai railway track and has a sprawling complex with gardens and greenery all around.

**About the department:** The Computer Science and Engineering department was started in the year 1984 with an intake of 40 students for UG. The department has seen phenomenal growth and now the department has increased UG intake to 120 students and offering two Post Graduation programmes : PG (Computer Science and Engineering with an intake of 25 students) and PG(Computer Network and Engineering with an intake of 18 students). The department is offering research program under its recognized research center. Computer Science and Design course was started from 2021 with an intake of 60 students. The department is having state-of-the-art computing facilities with high speed internet facilities and laboratories. The department library provides useful resources like books and journals. The department has well qualified and experienced teaching faculty. The department has been conducting several faculty development programs and student training programs.

## Vision of the Institution

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

## Mission of the Institution

- To provide a high quality educational experience for students with values and ethics thatenables them to become leaders in their chosen professions.
- To explore, create and develop innovations in engineering and science through research anddevelopment activities.
- To provide beneficial service to the national and multinational industries and communities through educational, technical, and professional activities

## Vision of the Department

- To become a premier department in Computer education, research and to prepare highlycompetent IT professionals to serve industry and society at local and global levels.

## Mission of the Department

- To impart high quality professional education to become a leader in Computer scienceand Engineering.
- To achieve excellence in Research for contributing to the development of the society.
- To inculcate professional and ethical behaviour to serve the industry.

**Program Educational Objectives (PEO)**:

| | |
|---|---|
| **PEO1:** | To prepare graduates with core competencies in mathematical and engineering fundamentals to solve and analyze computer science and engineering problems. |
| **PEO2:** | To adapt to evolving technologies and tools for serving the society. |
| **PEO3:** | To perform as team leader, effective communicator and socially responsible computer professional in multidisciplinary fields following ethical values |
| **PEO4:** | To encourage students to pursue higher studies, engage in research and to become entrepreneurs. |

**Program Outcomes:**

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

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

03. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

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

05. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities withan understanding of the limitations.

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

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

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

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

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

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

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

**Program Specific Outcomes (PSOs):**

| | |
|---|---|
| **PSO1:** | Acquire competency in hardware and software working principles to analyze and solve computing problems. |
| **PSO2:** | Design quality software to develop scientific and business applications following Software Engineering practices. |
| **PSO3:** | Apply cutting edge technologies using modern tools to find novel solutions ethically to existing problems. |

## SCHEME OF TEACHING FOR VI SEMESTER - 2022-2023

| Code No. | Course | Teaching Hours/Week | | | | Examination | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Lecture | Tutorial | Self Study | Practical | CIE | SEE | Total | Credits |
| 19HU61 | Entrepreneurship, Management and Finance | 3 | 0 | 0 | 0 | 50 | 50 | 100 | 3 |
| 19CS62 | Computer Networks | 3 | 0 | 0 | 0 | 50 | 50 | 100 | 3 |
| 19CS63 | Machine Learning | 4 | 0 | 1 | 0 | 50 | 50 | 100 | 4 |
| 19CS64X | Professional Elective- I | 3 | 0 | 0 | 0 | 50 | 50 | 100 | 3 |
| 19CS65X | Industrial Elective | 3 | 0 | 0 | 0 | 50 | 50 | 100 | 3 |
| 19CS6OE | Open Elective- I | 3 | 0 | 0 | 0 | 50 | 50 | 100 | 3 |
| 19HU02 | Recruitment Process Training-II | 2 | 0 | 0 | 0 | 50 | 50 | 100 | 1 |
| 19CSL61 | Computer Networks Lab | 0 | 0 | 0 | 2 | 50 | 50 | 100 | 1 |
| 19CSL62 | Machine Learning Lab | 0 | 0 | 0 | 2 | 50 | 50 | 100 | 1 |
| 19CSMP63 | Mini-project | 0 | 0 | 0 | 2 | 50 | 50 | 100 | 2 |
| | **Total** | **21** | **0** | **1** | **6** | **500** | **500** | **1000** | **24** |

### ELECTIVES OFFERED:

| **Professional Elective- I** | |
|---|---|
| 19CS641 | Internet of Things |
| 19CS642 | Advanced Data Structures |
| 19CS643 | Parallel Computing |
| 19CS644 | Digital Image Processing |

**Industry Elective**

| | |
|---|---|
| 19CS651 | Building Enterprise Applications |
| 19CS652 | Web Services |
| 19CS653 | Data Analytics |
| 19CS654 | Robotic Process Automation |

**Open Elective -1**

| | |
|---|---|
| 19CS6OE | Java Programming |

**AUTONOMOUS SYLLABUS FOR B.E VI SEMESTER 2022-2023**

| **Course Title** : **ENTREPRENEURSHIP, MANAGEMENT AND FINANCE** | | |
|---|---|---|
| **Subject Code :19HU61** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |

**Prerequisites:**

**Course Objectives :**

To enable the students to obtain the basic knowledge about Entrepreneurship and Management and finance in the following topics:-

- The Meaning, Functions, Characteristics, Types, Role and Barriers of Entrepreneurship,. Government Support for Entrepreneurship
- Management – Meaning, nature, characteristics, scope , functions, role etc and Engineers social responsibility and ethics
- Preparation of Project and Source of Finance
- Fundamentals of Financial Accounting
- Personnel and Material Management, Inventory Control

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**ENTREPRENEUR :** Meaning of Entrepreneur; Functions of an Entrepreneur; Characteristics of an entrepreneur , Types of Entrepreneur; Intrapreneurs – an emerging class ; Role of Entrepreneurs in economic development; Barriers to entrepreneurship, Government Support for Innovation and Entrepreneurship in India - Startup-India, Make-in-India, PMMY, AIM , STEP, BIRAC, Stand-up India, TREAD | **8 Hrs** |
| **Module II**<br>**MANAGEMENT:** Introduction – Meaning – nature and characteristics of Management, Scope and functional areas of management, Roles of Management, Levels of Management, Henry Fayol - 14 Principles to Management , Engineers Social responsibility and Ethics | **8 Hrs** |
| **Module III**<br>**PREPARATION OF PROJECT AND SOURCE OF FINANCE:**<br>**PREPARATION OF PROJECT:** Meaning of project; Project Identification; Project Selection; Project Report; Need and Significance of Report; Contents;<br>**SOURCE OF FINANCE**: Long Term Sources(Equity, Preference, Debt Capital, Debentures, loan from Financial Institutions etc) and Short Term Source(Loan from commercial banks, Trade Credit, Customer Advances etc) | **8 Hrs** |
| **Module IV**<br>**FUNDAMENTALS OF FINANCIAL ACCOUNTING:** Definition, Scope and Functions of Accounting , Accounting Concepts and Conventions: Golden rules of Accounting, Final Accounts - Trading and Profit and Loss Account, Balance sheet | **9 Hrs** |

| **Module V**<br>**PERSONNEL MANAGEMENT, MATERIAL MANAGEMENT AND INVENTORY CONTROL:**<br><br>**PERSONNEL MANAGEMENT**: Functions of Personnel Management, Recruitment, Selection and Training, Wages, Salary and Incentives<br>**MATERIAL MANAGEMENT AND INVENTORY CONTROL:** Meaning, Scope and Objects of Material Management. Inventory Control- Meaning and Functions of Inventory control ; Economic Order Quantity(EOQ) and various stock level ( Re-order level, Minimum level, Maximum level, Average level and Danger level) | **9 Hrs** |
|---|---|

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19HU61 | CO1 | Develop Entrepreneurship skills |
| | CO2 | Apply the concepts of management and Engineers Social responsibility & Ethics practice |
| | CO3 | Prepare project report & choose different Source of Finance. |
| | CO4 | Apply Fundamentals of Financial Accounting and interpret the final accounts |
| | CO5 | Apply personnel management skills, Material and inventory control techniques |

| **Course Title: COMPUTER NETWORKS** | | |
|---|---|---|
| Subject Code : **19CS62** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** Data Communication**.** | | |

**Course Objectives:**

- Review the underlying concepts of data communication and computer network
- Gain knowledge about functions of data link layer and related protocols and Describe working of simple LAN with hubs, bridges and switches
- Learn Internetworking with emphasis on routing protocols, architectures, and implementation issues
- Learn functions of transport layer, study congestion control mechanisms and build network applications using TCP/IP model.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Review of Basic Concepts& Direct Link Networks:** Building a Network; Requirements-Connectivity, Cost-Effective Resource Sharing, support for Common Services; Network Architecture-Layering and Protocols, OSI Architecture, Internet Architecture; Performance-Bandwidth and Latency, Delay × Bandwidth Product, High-Speed Networks. ; Encoding (NRZ, NRZI, Manchester, 4B / 5B), Framing: (BISYNC,PPP,DDCMP),HDLC,SONET | **08 Hrs** |
| **Module II**<br>**Packet Switching:** Two dimensional parity, internet checksum algorithm, CRC, Reliable transmission: stop and wait, sliding window, concurrent logical channels Ethernet & multiple access networks(802.3): physical properties, access protocol, experience with Ethernet,Wireless:802.11/Wi-Fi, Bluetooth , cell phone technologies, Switching and Bridging –Datagrams , Virtual Circuit Switching, Source Routing; Bridges and LAN Switches. | **08 Hrs** |
| **Module III**<br>**Internetworking :** Basic Internetworking: What is an internetwork, service model, global addresses, datagram forwarding in IP, subnetting & classless addressing Address Translation(ARP), Host configuration(DHCP),Error Reporting(ICMP), Virtual Networks and Tunnels, Routing**,** Network as a Graph, distance Vector(RIP), Link State(OSPF), Metrics, Global Internet Routing areas, Interdomain routing (BGP),IP Version 6(IPv6). | **08 Hrs** |

| | |
|---|---|
| **Module IV**<br>**End–to-End Protocols& Resource Allocation:** Simple demultiplexer(UDP); Reliable byte stream(TCP)–End-to-End Issues, Segment Format, Connection Establishment and Termination, Sliding Window Revisited, Triggering Transmission, Adaptive Retransmission, record Boundaries, Issues in resource allocation–Network Model, Taxonomy, Evaluation Criteria; Queuing disciplines. | **08 Hrs** |
| **Module V**<br>**Congestion Control & Application:** TCP Congestion Control–Additive Increase/Multiplicative Decrease, Slow Start, Fast Retransmit and Fast Recovery; Congestion-Avoidance mechanisms–DECbit, Random Early Detection(RED), Source-Based Congestion Control. Traditional applications–Electronic Mail (SMTP, MIME, IMAP), World Wide Web(HTTP),<br>Multimedia application**:** session and call control Infrastructure Services:Name Service (DNS), Network management (SNMP) Quality of Service. | **10 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**

1. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 5th Edition, Elsevier, 2010.

**REFERENCE BOOKS:**

1. Behrouz A. Forouzan: Data Communications and Networking, 4th Edition, Tata McGraw Hill, 2006.
2. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
3. Alberto Leon-Garcia and Indra Widjaja: Communication Networks -Fundamental Concepts and Key Architectures, 2nd Edition Tata McGraw-Hill, 2004.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS62** | **CO1** | Explain architectural concepts of layering , error control techniques and analyze data link protocols their analysis |
| | **CO2** | Illustrate the working of LAN , bridges and switches |
| | **CO3** | Apply principles of internetworking and illustrate internet routing protocols |
| | **CO4** | Demonstrate working of transport layer protocols and resource allocation |
| | **CO5** | Explore TCP congestion control techniques and application layer protocols |

| **Course Title: MACHINE LEARNING** | | | |
|---|---|---|---|
| Subject Code : **19CS63** | Credits :**4** | | CIE: 50 |
| Number of Lecture Hours/Week | **4 Hrs** | **Self-Study : 1Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 52 | | SEE Hours: 03 |

**Prerequisite:** Probability and statistics, knowledge of any programming language.

**Course Objectives:**

- Acquiring the fundamentals concepts machine learning
- Study various learning methods to develop an intelligent machine.

| **Modules** | **Teaching Hours** |
|---|---|
| **Module – I**<br>**Introduction:** Well posed learning problems, Designing a Learning system, Perspective and Issues in Machine Learning.<br>**Concept Learning:** Concept learning task, Concept learning as search, Find-S algorithm, Version space, Candidate Elimination algorithm, Inductive Bias. | **10 Hrs** |
| **Module - II**<br>**Decision Tree Learning:** Decision tree representation, Appropriate problems for decision tree learning, Basic decision tree learning algorithm, hypothesis space searching decision tree learning, Inductive bias in decision tree learning, Issues in decision tree learning. | **10 Hrs** |
| **Module - III**<br>**Artificial Neural Networks:** Introduction, Neural Network representation, Appropriate problems, Perceptron, Multilayer networks and the Back-propagation algorithm. | **10 Hrs** |
| **Module - IV**<br>**Evaluating Hypothesis:** Motivation, Estimating hypothesis accuracy, Basics of sampling theorem, General approach for deriving confidence intervals, Difference in error of two hypothesis, Comparing learning algorithms.<br>**Reinforcement Learning**: Introduction, The Learning Task, Q Learning, Nondeterministic Rewards and Action, Temporal Difference Learning.<br>**Instance Based Learning:** Introduction, k-nearest neighbor learning, locally weighted regression, radial basis function, cased-based reasoning. | **11 Hrs** |
| **Module – V**<br>**Bayesian Learning:** Introduction, Baye‟s theorem, Bayes theorem and concept learning, ML and LS error hypothesis, ML for predicting probabilities, MDL principle, Naive Bayes classifier, Bayesian belief networks, EM algorithm. | **11 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text book:**

1. Tom M. Mitchell, Machine Learning, Indian Edition Paperback 2017, McGraw Hill Education.

**Reference Books:**

1. Trevor , The Elements of Statistical Learning, 2nd edition, 2017, Springer series in statistics. Hastie, Robert Tibshirani, Jerome Friedman
2. Ethem Alpaydın, "**Introduction to machine learning"**, Third Edition, PHI Learning Pvt. Ltd. 2016

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS63 | **CO1** | Identify the problems for machine learning. |
| | **CO2** | Implement decision algorithm for machine learning problems |
| | **CO3** | Apply concepts of ANN for solving machine learning problems |
| | **CO4** | Evaluate hypothesis , use Reinforcement and instance based learning |
| | **CO5** | Illustrate Bayeians learning laws and its applications. |

| **Course Title: Internet of Things (IoT)** | | |
|---|---|---|
| Subject Code : **19CS641** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** Microprocessor and microcontroller | | |
| **Course Objectives:**<br>• Acquire the data with sensors and perform data analysis | | |

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>What is IoT, Genesis of IoT, IoT and Digitization, IoT Impact, Convergence of IT and IoT, IoT Challenges, IoT Network Architecture and Design, Drivers Behind New Network Architectures, Comparing IoT Architectures, A Simplified IoT Architecture, The Core IoT Functional Stack, IoT Data Management and Compute Stack. | **8 Hrs** |
| **Module II**<br>Smart Objects: The "Things" in IoT, Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects, Communications Criteria, IoT Access Technologies. | **8** Hrs |
| **Module III**<br>IP as the IoT Network Layer, The Business Case for IP, The need for Optimization, Optimizing IP for IoT, Profiles and Compliances, Application Protocols for IoT, The Transport Layer, IoT Application Transport Methods. | **8** Hrs |
| **Module IV**<br>Data and Analytics for IoT, An Introduction to Data Analytics for IoT, Machine Learning, Big Data Analytics Tools and Technology, Edge Streaming Analytics, Network Analytics, Securing IoT, IOT AND M2M,Introduction, difference between IoT and m2m,SDN and NFV for IOT. IoT Physical Devices and Endpoints - Arduino UNO: Introduction to Arduino, Arduino UNO, Installing the Software, Fundamentals of Arduino Programming. | **8 Hrs** |
| **Module V**<br>IoT Physical Devices and Endpoints - RaspberryPi: Introduction to RaspberryPi, About the RaspberryPi Board: Hardware Layout, Operating Systems on RaspberryPi, Configuring RaspberryPi, Programming RaspberryPi with Python, Wireless Temperature Monitoring System Using Pi, DS18B20 Temperature Sensor, Connecting Raspberry Pi via SSH, Accessing | **10 Hrs** |

| | |
|---|---|
| Temperature from DS18B20 sensors, Remote access to RaspberryPi, Smart and Connected Cities, An IoT Strategy for Smarter Cities, Smart City IoT Architecture, | |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books:**

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry,"IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 1st Edition, Pearson.
2. Srinivasa K G, "Internet of Things", CENGAGE Leaning India, 2017 .
3. Internet Of Things A hands on Approach , Arashdeep Bhaga,Vijay Madiseeti

**Reference Books:**

1. Vijay Madisetti and ArshdeepBahga, "Internet of Things (A Hands -on-Approach)", 1 st Edition, VPT, 2014.
2. Raj Kamal, "Internet of Things: Architecture and Design Principles", 1 st Edition, McGraw Hill Education, 2017.

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS641 | CO1 | Illustrate the impact and challenges posed by IoT networks leading to new architectural models. |
| | CO2 | Compare and contrast the deployment of smart objects and the technologies to connect them to network. |
| | CO3 | Demonstrate the role of IoT protocols for efficient network communication. |
| | CO4 | Describe the need for Data Analytics and Security in IoT. |
| | CO5 | Analyze different sensor technologies for sensing real world entities and identify the applications of IoT in Industry. |

| Course Title: **ADVANCED DATA STRUCTURES** | | |
|---|---|---|
| Subject Code : **19CS642** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |

**Pre-requisites:** Data structures, Analysis and Design of Algorithms**.**

**Course objectives:**

- Understand advanced data structures to solve complex problems in various domains.
- Build the logic to use appropriate data structure in logical and computational solutions.
- Develop modern applications using Advanced data structures.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module –I**<br><br>**Introduction:** Data Structure, Design of Data Structure, Analysis of Data Structure,<br>Amortized Complexity, Computational Models, RAM model, Word RAM model, Cell-probe model of computation, Bounds of Fundamental Data Structures, Lazy Delete, **O(1) Search by Hashing** : Basic Hashing, Perfect Hashing, Universal Hashing, Cuckoo Hashing, Bloom Filters, Locality-Sensitive Hashing.<br>**O(log(n)) Ordered Search (Trees and Lists):** Balanced Binary Search Trees (BSTs), Randomized BSTs, Splay Tree, Tango Tree, Skiplists, Static and Dynamic Optimality. | **08 Hrs** |
| **Module-II**<br><br>**Findset, Find Min, and Find Word:** Disjoint Sets, Binomial Heap, Fibonacci Heaps, Tries, Inverted Index.<br>**Evolving Paradigms of Data Structures:** Geometric Queries, I/O Complexities, Communication Complexities, Large Data Problem. | **08 Hrs** |
| **Module-III**<br><br>**Spatial Data Structures:** Range Search Trees, KD Trees, Quadtree, R Tree.<br>**Temporal Data Structures:** Partial Persistence, Retroactivity.<br>**External Memory Data Structures:** Input/Output (I/O) Model, Cache Oblivious Algorithms, B, B+ Tree, (a,b) Tree, Buffer Tree. | **09 Hrs** |
| **Module- IV**<br>**Distributed Data Structures (DDSs):** Descriptions of Structures, Distributed Hashing, Distributed Trees, Skip Graphs.<br>**Synopsis Data Structures:** Data Synopsis, Sampling, Sketching, Fingerprint, Wavelets. | **08 hrs** |
| **Module – V**<br><br>**Recent Applications: Introduction to Applications:** Various Domain Applications, Project.<br>**Applications to Cryptography :** MD5, Secure Socket Layers (SSLs), Block Chains, Digital Signature, Projects. | |

| | |
|---|---|
| **Application to IR and WWW**: Crawl Frontier, Posting List Intersection, Text Retrieval from Inverted Index, Auto Complete Using Tries, Projects.<br>**Applications to Data Science:** Heavy Hitters and Count-Min Structures, Approximate Nearest Neighbor Searches, Low Rank Approximation by Sampling, Near-Duplicate Detection by Min Hashing. Projects.<br>**Application to Network and IOT:** Click-Stream Processing Using Bloom Filters, GBF Algorithm, Fast IP-Address Lookup Using Tries, Integrity Verification: Cloud and IOT Data, Projects.<br>**Applications to Systems:** Queue Spilling, Completely Fair Schedulers in Kernels, CFS internals, Distributed Caching, Data Structures for Building File Systems, Projects.<br>Applications to Databases: Database Problems, B and B+ Trees for Database Creation and Block Search, CouchDB, Bloomjoins. Projects.<br>**Applications to Images and Graphics:** R Trees for Map Searches, Spatial Proximity in GIS, Ray Shooting, Data Structures Used in Ray Shooting.<br>Projects. | **09 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text books:**

1. Suman saha, Shailendra shukla, "Advanced Data Structures Theory and Applications", CRC Press, Tailor and Francis, publication, Jun 2019, ISBN: 9781138592605
2. https://www.tutorialspoint.com/advanced_data_structures

**Reference Books:**

1. Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford *." Introduction to Algorithms (3rd ed.)." MIT Press and McGraw-Hill.ISBN 0-262-03384-4. 1320* pp., 5 printings up to 2016
2. Peter Brass, "Advanced data structures", Cambridge University Press, 1st edition, sep 2008, 2008,ISBN: 978-0521880374
3. Marcello Ra Rocca, "Advanced algorithm and data structures", Manning publications, May 2021

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS642** | **CO1** | Demonstrate, design and analysis of data structures, hashing, ordered search. |
| | **CO2** | Illustrate heaps, tries and I/O complexities. |
| | **CO3** | Apply spatial, temporal and external memory data structures to large volume of data. |
| | **CO4** | Demonstrate Distributed and synopsis data structures . |
| | **CO5** | Develop various domains real world applications . |

| Course Title: **PARALLEL COMPUTING** | | |
|---|---|---|
| Subject Code : **19CS643** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Pre-requisite:** Data Structure, Analysis and design of algorithms, Operating System | | |
| **Course Objectives:**<br>• To introduce the major concepts and ideas in parallel computing and its applications<br>• To help understand various model of parallelism and their strength and limitations.<br>• To introduce bottle necks in parallel computing<br>• Basic knowledge to write simple MPI parallel programs. | | |
| **MODULES** | | **Teaching Hours** |
| **Module –I**<br><br>**Introduction to Parallel Computing & Parallel Programming Platforms:** Motivating Parallelism, Scope of Parallel Computing, Implicit Parallelism: Trends in Microprocessor Architectures, Limitations of Memory System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines, Routing Mechanisms for Interconnection Networks, Impact of Process-Processor Mapping and Mapping Techniques. | | **08 Hrs** |
| **Module-II**<br><br>**Principles of Parallel Algorithm Design: Preliminaries**, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models. | | **08 Hrs** |
| **Module-III**<br><br>**Analytical Modeling of Parallel Programs :** Sources of Overhead in Parallel Programs, Performance metrics for parallel systems, The effect of Granularity on performance, Scalability of Parallel Systems, Minimum execution time and minimum cost optimal execution time, Asymptotic analysis of Parallel programs, Other Scalability Metrics. | | **08 Hrs** |
| **Module- IV**<br><br>**Programming Using the Message Passing Paradigm:** Principles of Message – Passing Programming, The Building Blocks, MPI: The Message passing Interface, Overlapping Communication with Computation, Collective Communication and Computation Operations, Groups & Communicators. | | **08 hrs** |
| **Module – V**<br><br>**Pthreads and Synchronization:** Thread Basics, POSIX Thread API, Synchronization Primitives in Pthreads,Controlling Thread and | | |

| | |
|---|---|
| Synchronization Attributes, Thread Cancellation, Composite Synchronization Constructs.<br><br>**OpenMP :** Open MP programming model, Specifying tasks in openMP, Synchronization constructs in open MP, Data handling in OpenMP, Open MP library functions, Environment variables in OpenMP, Explicit Thread versus OpenMP based programming. . | **10 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text books:**

1. Ananth Grama, George Karypis, Vipin Kumar and Anshul Gupta, Introduction to Parallel Computing, Second Edition, Pearson Education, 2014

**Reference Books:**

1. Michael Quinn, Parallel Computing Theory and Practice, Tata McGraw Hill, 2003.
2. Michael Quinn, Parallel Programming in C with Mpi and openMP, McGrawHill, 2017
3. Peters S Pacheco, An introduction to Parallel Programming, Morgan Kaufmann Publishers

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS643** | **CO1** | Explain Parallel computing, parallel programming Platforms, its scope and routing mechanisms for interconnection networks |
| | **CO2** | Apply Principles of Parallel Algorithm Design through use of mapping techniques for load balancing |
| | **CO3** | Analyze Parallel Programs and evaluate various performance metrics Granularity, Scalability, execution time and cost |
| | **CO4** | Discuss message Passing interface, Communication and Computation Operations. |
| | **CO5** | Illustrate parallel programming through implementation of Pthreads and Synchronization and describe the OpenMP programming model**.** |

| Course Title: **DIGITAL IMAGE PROCESSING** | | |
|---|---|---|
| Subject Code : **19CS644** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |

**Pre-requisite:** Mathematics

**Course Objectives:**

- To comprehend the relation between human visual system and machine perception and processing of digital images.
- To provide a detailed approach towards image processing applications like enhancement, segmentation, and compression.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module –I**<br>**Introduction :** Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Sampling and Quantization, Representing Digital Images (Data structure), Some Basic Relationships Between Pixels- Neighbors and Connectivity of pixels in image, Applications of Image Processing: Medical imaging, Robot vision, Character recognition, Remote Sensing. | **09Hrs** |
| **Module-II**<br>**Image Enhancement In The Spatial Domain:** Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods. | **08 Hrs** |
| **Module-III**<br>**Image Segmentation:** Introduction, Detection of isolated points, line detection, Edge detection, Edge linking, Region based segmentation- Region growing, split and merge technique, local processing, regional processing, Hough transform, Segmentation using Threshold | **08 Hrs** |
| **Module- IV**<br>**Image Compression:** Introduction, coding Redundancy , Inter-pixel redundancy, image compression model, Lossy and Lossless compression, Huffman Coding, Arithmetic Coding, LZW coding, Transform Coding, Sub-image size selection, blocking, DCT implementation using FFT, Run length coding. | **09 hrs** |
| **Module – V**<br>**Representation and Description:** Representation, Boundary descriptors. Region descriptions. Application using MATLAB or Python on Shape and texture descriptors | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books:**

- Digital Image Processing – Rafael C. Gonzalez, Richard E. Woods, 3rd Edition, Pearson, 2008
- Digital Image Processing- S Jayaraman, S Esakkirajan, T Veerakumar- MC GRAW HILL EDUCATION, 2010.

**Reference Books:**

- Digital Image Processing and Analysis-Human and Computer Vision Application with using CVIP Tools – Scotte Umbaugh, 2nd Ed, CRC Press, 2011
- Digital Image Processing using MATLAB – Rafael C. Gonzalez, Richard E Woods and Steven L. Eddings, 2nd Edition, MC GRAW HILL EDUCATION, 2010.
- Digital Image Processing and Computer Vision – Somka, Hlavac, Boyle- Cengage Learning (Indian edition) 2008.
- Introductory Computer Vision Imaging Techniques and Solutions- Adrian low, 2008, 2nd Edition

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS644 | **CO1** | Explore the fundamentals of digital images and its representations |
| | **CO2** | Elaborate understanding on image enhancement techniques. |
| | **CO3** | Apply and Discuss image segmentation techniques |
| | **CO4** | Justify the need for compression and evaluate the basic compression algorithms. |
| | **CO5** | Implement boundary and texture descriptors |

| Course Title**: BUILDING ENTERPRISE APPLICATIONS** | | |
|---|---|---|
| Subject Code : **19CS651** | Credits :3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Pre-requisites:** Basics of software Engineering. | | |

**Course Objectives:**

- To familiarize with different application frameworks.
- To acquire knowledge about software architecture and design at enterprise level
- To get practical aspects of planning and estimation in enterprise application rollout.
- To get acquainted with code review, code analysis and build process.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Introduction** to enterprise applications and their types, software engineering methodologies, life cycle of raising an enterprise application, introduction to skills required to build an enterprise application, key determinants of successful enterprise applications, and measuring the success of enterprise applications.<br><br>Inception of enterprise applications, enterprise analysis, business modeling, requirements elicitation, use case modeling, prototyping, non functional requirements, requirements validation, planning and estimation.<br><br>**Architecture, Views and Viewpoints: Enterprise Application:** An Enterprise Architecture Perspective, Enterprise Triangle and Enterprise Architecture, Enterprise Architecture frameworks, Blueprint of an Enterprise Application. | **09 Hrs** |
| **Module II**<br>**Logical Architecture: Technical Architecture and Design:** Mapping Logical Architecture to Technical Architecture, Object-Oriented Analysis and Design, Infrastructure Services Layer, Presentation Layer, Business Layer, External Systems Layer, Integration Layer, Technical Solution Ecosystem. | **08 Hrs** |
| **Module III**<br>**Data Architecture Design:** Relational Data modeling, XML modeling , other Structured data representation, unstructured data representation . **Infrastructure Architecture and Design:** Infrastructure Architecture and Design, Networking, Internetworking , Internetworking and Communication Protocols, IT Hardware and Software, Middleware, Policies for Infrastructure Management Deployment Strategy, **Architecture and Design Documentation:** System Architecture Documentation, System Architecture Documentation , Design Documentation. **Construction Readiness:** Defining a Construction Plan, Defining a Package Structure, Setting Up a Configuration Management Plan, Setting Up a Development Environment. | **08 Hrs** |

| | |
|---|---|
| **Module IV**<br>**Introduction to Software Construction Map, Constructing the Solution Layers:** Infrastructure Services Layer Components, Presentation Layer Components, Business Layer Components, Data Access Layer Components, Integration Layer Components.<br>**Code Review**: Objectives, Process. **Static Code Analysis**: Coding Style, Logical Bugs, Security Vulnerabilities, Code Quality. **Building Process and Unit Testing:** Building Process, Unit Testing. **Dynamic Code Analysis:** Code Profiling, Code Coverage. | **08 Hrs** |
| **Module V**<br>**Testing and Rolling Out Enterprise Applications: Testing Enterprise Applications:** Types and Methods of Testing, Testing Levels, Testing Approach, Enterprise Application Environments, Integration Testing.<br><br>**System Testing:** Performance Testing, Penetration Testing Usability Testing, Usability Testing, Globalization Testing, And Interface Testing. User Acceptance Testing. Rolling Out Enterprise Applications. | **09 Hrs** |

**Question paper pattern:**

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

**Text book:**

1. Raising Enterprise Applications, Anubhav, Pradhan, Satheesha B Nanjappa, Senthi K Nallasamy, Veerakumar Esakimuthu,, John Wiley, 2015

**Reference Books:**

1. Software Requirements: Styles & Techniques – published by Addison – Wesley Professional.
2. Software Systems Requirements Engineering in Practice – published by McGraw-Hill/Osborne Media.
3. Managing Software Requirements: A use Case Approach, 2/e – published by person.
4. Software Architecture : A Case based Approach – published by Pearson
5. Building Java Enterprise Applications – Published by O,,Reilly Media authored by Brett McLaughlin

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS651 | CO1 | Illustrate Enterprise Applications and its architecture. |
| | CO2 | Demonstrate the design of Logical and Technical architecture. |
| | CO3 | Discuss design of data and infrastructure architecture. |
| | CO4 | Develop Construction plan and Perform Code review, Code analysis. |
| | CO5 | Employ testing methods and develop applications through enterprise concepts |

| **Course Title: WEB SERVICES** | | |
|---|---|---|
| SubjectCode:**19CS652** | Credits:3 | CIE:50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE:50 |
| Total Number of Lecture Hours | 42 | SEEHours:03 |
| **Pre-Requisite: Web Technology** | | |
| **Course objectives:**<br>• Define and explain Web Services and its architectures..<br>• Gain working knowledge of designing scalable web services. | | |
| **MODULE** | | **Teaching Hours** |
| **MODULE-I**<br>**The Programmable Web and Its Inhabitants:** Kinds of Things on the Programmable Web4 HTTP: Documents in Envelopes, Method Information, Scoping Information, The Competing Architectures, Technologies on the Programmable Web, Leftover Terminology, Writing Web Service Clients: Web Services Are Web Sites, delicious, The Sample Application Making the Request: HTTP Libraries, Processing the Response: XML Parsers, JSON Parsers: Handling Serialized Data, Clients Made Easy with WADL | | **09 Hrs** |
| **Module– II**<br>RESTful Services: Introducing the Simple Storage Service Object-Oriented Design of S3, Resources, HTTP Response Codes, An S3 Client, Request Signing and Access Control, Using the S3 Client Library, Clients Made Transparent with Active Resource, Parting Words, The Resource-Oriented Architecture, Resource-Oriented URIs, Addressability, Statelessness, Representations, Links and Connectedness, The Uniform Interface, That‟s It! | | **08 Hrs** |
| **Module-III**<br>Designing Read-Only Resource-Oriented Services: Resource Design, Turning Requirements Into Read-Only Resources, Figure Out the Data Set, Split the Data Set into Resources, Name the Resources, Design Your Representations, Link the Resources to Each Other, The HTTP Response, Conclusion, Designing Read/Write Resource-Oriented Services, User Accounts as Resources, Custom Places, A Look Back at the Map Service. | | **08 Hrs** |
| **Module-IV**<br>A Service Implementation: A Social Bookmarking Web Service, Figuring Out the Data Set, Resource Design, Design the Representation(s) Accepted from the Client, Design the Representation(s) Served to the Client, Connect Resources to Each Other, What‟s Supposed to Happen, What Might Go Wrong? ,Controller Code, Model Code, What Does the Client Need to Know? , REST and ROA Best Practices :Resource-Oriented Basics, The Generic ROA Procedure, Addressability, State and Statelessness, Connectedness , The Uniform Interface, This Stuff Matters, Resource Design, URI Design, Outgoing Representations, Incoming Representations, Service Versioning, Permanent URIs Versus Readable URIs, Standard Features of HTTP, Faking PUT and DELETE, The Trouble with Cookies, Why Should a User Trust the HTTP Client. | | **09 Hrs** |

| **Module-V** | |
|---|---|
| The Building Blocks of Services: Representation Formats, Prepackaged Control Flows, Hypermedia Technologies, The Resource-Oriented Architecture Versus Big Web Services, What Problems Are Big Web Services Trying to Solve? ,SOAP, WSDL, UDDI, Security, Reliable Messaging, Transactions, BPEL, ESB, and SOA, Conclusion. | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be questions from each module, covering all the topics from a module.
The students will have to answer 5fullquestions, selecting one full question from each module**.**

**Text Books:**
1. Leonard Richardson and Sam Ruby Beijing Cambridge Farnham Köln Sebastopol Tokyo "RESTful Web Services", First Edition ,Copyright © 2007 O"Reilly Media. Printed in the United States of America. Published by O"Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol.
**Refer the above mentioned text book for Module 1,2,3, 4 & 5**

**Reference Books:**
1. RESTful Java with JAX-RS 2.0*:* Designing and Developing Distributed Web Services *2nd Edition, ISBN-13978-1449361341,Edition,PublisherO'Reilly Media, Publication date12 November 2013.*

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

| Course Code | CO# | Course Outcome(CO) |
|---|---|---|
| 19CS652 | **CO1** | Identify the fundamentals of Web service architectures |
| | **CO2** | Explore the RESTful Services. |
| | **CO3** | Design the Read-Only Resource-Oriented Services |
| | **CO4** | Implementation of web Services and best practices. |
| | **CO5** | Illustrate Building Blocks Of Web Services |

| **Course Title: DATA ANALYTICS** | | |
|---|---|---|
| **Subject Code :19CS653** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites: Data Base Management System** | | |
| **Course Objectives :** Introduce student to data analytic on distributed platform | | |

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Big data processing and Distributed architectures** -Types of data: Structured, semi structured, unstructured **,** Data Pre-processing: Data cleaning, Data Integration, Data Reduction, Data Transformation and discretization, data cleaning, validation, modifications, enhancements.<br>Distributed Architectures : Hadoop, spark, HPCC Systems VsHadoop | **08 Hrs** |
| **Module II**<br>**HPCC Systems architecture :**HPCC System functions, Data Lake Architecture, The HPCC Systems design, Thor Vs ROXIE<br>**ECL the programming language &Structures :**ECL Watch, ECL Cloud IDE / VS Code, Simple ECL programs and Data Types explained, Data flow graphs (diagrams), Declarative programming, Declarative vs Imperative programming, the ECL Compiler, The ECL program deployment and execution. | **09 Hrs** |
| **Module III**<br>**ECL the programming language &Structures :**An Activity, An Activity Declaration, A Record Declaration, Schema on Read (RECORD) explained, A Function Declaration, A MODULE, ECL File(s), Importing files, Spraying and Reading a file<br>**Data Shaping (Transforming) :**Function, Module And Project, Iterate And Rollup ,Sort, Join And Dedup ,Normalize And Denormalize ,Distribute And Reading The Execution Graph | **09 Hrs** |
| **Module IV**<br>**Data Aggregation**<br>GROUP and functions (SUM, AVE, COUNT…), TABLE and AGGREGATE<br>**HPCC Systems Machine Learning Library- Part I**<br>ML_Core, PBblas- Parallel Block Linear Algebra Subsystem, Supervised Learning Bundles- Linear Regression, Logistic Regression, Support Vector Machines, Learning Trees | **08 Hrs** |
| **Module V**<br>**HPCC Systems Machine Learning Library- Part II**<br>Supervised Learning Bundles- GLM, Generalized Neural Network, Unsupervised Learning Bundles- K-Means, DBSCAN, Natural Language Processing Bundles-TextVectors | **08 Hrs** |

**Question paper pattern**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.

The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**

1. Big Data and Analytics, Seema Acharya and Subhashini C, 1st Edition Wiley India Private Limited, 2015,ISBN 978-8126554782. Module .

Detailed handouts with references to material available on the web will be handed out every week.

https://hpccsystems.com/training/documentation/learning-ecl

https://github.com/hpcc-systems/Solutions-ECL-Training, Module 2 and 3

2. Data Mining – Concepts and Techniques, Jiawei Han and Micheline Kamber, Jian Pei, 3rd Edition,Morgan Kaufmann, 2012, ISBN 978-0-12-381479-1. module 1, module 4 & 5
3. Introduction to Data Mining, Pang-Ning Tan, Michael Steinbach, Vipin Kumar:, Pearson Education, 2007, ISBN 978-81-317-1472-0.

**REFERENCE BOOKS:**

1. Paulraj Ponnaiah John Wiley & Sons, "Data Warehousing Fundamentals –Inc ", Student Edition, 2001.
2. Margaret H Dunham, "Data Mining Introductory and advanced topics" –Pearson education, 2003.
3. Arun K Pujari, "Data Mining Techniques" –University Press, Private Limited, 2013.
4. C.C. Aggarwal, "Data Mining" Springer International Publishing Switzerland 2016.
5. Douglas Eadline,"Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem", 1stEdition, Pearson Education, 2016. ISBN: 978- 9332570351.
6. Joey Echeverria, Ben Spivey, "Hadoop Security", O'Reilly Media, Inc., *2016*.ISBN: 981491900987.

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS653 | CO1 | Understand and explore the concepts of data processing, distributed systems |
| | CO2 | Explore HPCC systems, ECL processing languages and structures |
| | CO3 | Apply ECL processing and structure and process of data shaping |
| | CO4 | Describe data and analysis and machine algorithm on HPCC platform |
| | CO5 | Implement HPCC systems machine learning library |

| **Course Title: ROBOTIC PROCESS AUTOMATION** | | |
|---|---|---|
| Subject Code : **19CS654** | **Credit : 3** | CIE: 50 |
| Number of Lecture Hours/Week | **42 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 3 | SEE Hours: 03 |

**Prerequisites:** Problem Solving with Programming , Object Oriented Programming with Java and Operating Systems .

**Course Objectives:**

1. Discuss the concepts of Robotics Process automation
2. Describe the sequence, flowchart and control flow in automation tool
3. Demonstrate the data manipulation techniques
4. Demonstrate the usage of UI Explorer and Screen scraping.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**What Is Robotic Process Automation**: Scope and techniques of automation, Robotic process automation, About UiPath, Future of Automation. **Record and Play**: UiPath stack, Downloading and installing UiPath Studio, Learning UiPath Studio, Task recorder, Step-by-step examples using the recorder. | **9 Hrs** |
| **Module II**<br>**Sequence. Flowchart and Control Flow**:Sequencing the Workflow, Activities, Control Flow, various types of loops, and decision making, Step-by-step example using Sequence and Flowchart, Step-by-step example using Sequence and Control Flow | **9 Hrs** |
| **Module III**<br>**Data Manipulation:** Variables and Scope, Collections, Arguments-Purpose and use, Data table usage and examples, Clipboard management, File operation with step-by-step example, CSV/Excel to data table and vice versa with a step-by-step example | **8 Hrs** |
| **Module IV**<br>**Taking Control of the Controls:** Finding and attaching windows, Finding the control, Techniques for waiting for a control, Act on controls-mouse and keyboard activities | **8 Hrs** |
| **Module V**<br>**Working with UI Explorer**: Handling events, Screen Scraping, When to use OCR, Types of OCR available, How to use OCR. | **8 Hrs** |

**Question paper pattern:**

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

**TEXT BOOKS:**

1. Learning Robotic Process Automation Alok mani tripathi Kindle Edition, Published rch by Packt Publishing .
2.E. Turban, R. Sharda, D. Delen, David King, Business Intelligence, 2nd ed. Pearson India, 2010.

**REFERENCES:**

1. Marlon Dumas et. al., Fundamentals of Business Process Management, Springer, ebook, 2012.
2. Van der Aalst, Process Mining: Discovery, Conformance and Enhancement of Business Processes, Third edition, 2011.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS654 | CO1 | Identify the Robotics Process automation tools |
| | CO2 | Implement the sequence, flowchart and control flow in UiPath Studio |
| | CO3 | Implement the data manipulation techniques in UiPath Studio |
| | CO4 | Discuss the UI Explorer and Screen scraping techniques |
| | CO5 | Implement the concepts learnt for real world applications |

| **Course Title: JAVA PROGRAMMING** | | |
|---|---|---|
| Subject Code : **19CS6OE** | **Credit : 03** | CIE: 50 |
| Number of Lecture Hours/Week | **03 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** C programming | | |
| **Course Objectives:** Learn the Java Programming to develop applications, understanding concepts of multithreading and Exception Handling , creating GUI with applets and Event Handling. | | |
| **MODULES** | | **Teaching Hours** |
| **Module I**<br>**Java Programming fundamentals**– The Java Language, The Key attributes of Object-Oriented Programming, The Java Development Kit, Handling Syntax Errors, The Java Keywords, Identifiers in Java, The Java Class Libraries.<br>**Introducing Data Types and Operators:** Why Data types Are Important**,** Java Primitive Types , Literals, A Closer Look at Variables, The Scope and Lifetime of Variables, Operators, Arithmetic Operators, Relational and Logical Operators, Short-Circuit Logical Operators, The Assignment Operator, Shorthand Assignments, Type Conversion in Assignments, Using a Cast, Operator Precedence, Expressions.<br>**Control Statements, String Handling-** String Fundamentals, The String Constructors, Three String-Related Language Features, The Length() Method, Obtaining the Characters within a String, String Comparison, Using indexOf() and lastIndexOf(), Changing the Case of Characters Within a String. | | **10 Hrs** |
| **Module II**<br>**More Data Types and Operators** – Arrays, Multidimensional Arrays, Alternative Array Declaration Syntax, Assigning Array References, Using the Length Member, The For-Each Style for Loop, Strings, The Bitwise Operators.<br>**Introducing Classes, Objects, and Methods**-Class Fundamentals, How Objects are created, Reference Variables and Assignment, Methods, Returning from a Method, Returning a value, Using Parameters, Constructors, Parameterized Constructors, The new Operator Revisited, Garbage Collection and Finalizers, The this Keyword. | | **08 Hrs** |
| **Module III**<br>**A Closer Look at Methods and Classes -** Controlling Access to Class Members, Pass Objects to Methods, How Arguments are passed, Returning Objects, Method Overloading, Overloading Constructors, Recursion, Understanding Static, Introducing Nested and Inner Classes, Varargs<br>**Inheritance- :** Inheritance Basics, Member Access and Inheritance, Constructors and Inheritance, Using super to Call Superclass constructors, Using super to Access Superclass Members, Creating a Multilevel Hierarchy, When are Constructors Executed, Superclass References and Subclass Objects, Method Overriding, Overridden Methods support polymorphism, Using Abstract Classes, Using final, The Object Class. | | **08 Hrs** |
| **Module IV**<br>**Interfaces:** Interface Fundamentals, Creating an Interface, Implementing an Interface, Using Interface References Implementing Multiple Interfaces, Constants in Interfaces, Interfaces can be extended, Nested Interfaces, Final Thoughts on Interfaces.<br>**Packages: Packages:** Package Fundamentals, Packages and Member Access | | |

| | |
|---|---|
| Importing Packages, Static Import.<br>**Exception Handling :**The Exception Hierarchy, Exception Handling Fundamentals, The Consequences of an Uncaught Exception, Exceptions Enable you to handle errors gracefully, using Multiple catch clauses, Catching subclass Exceptions, try blocks can be nested, Throwing an Exception, A Closer look at Throwable, using finally, using throws, Java‟s Built-in Exception, New Exception features added by JDK 7, Creating Exception Subclasses. | **08 Hrs** |
| **Module V**<br>**Applets:** Applet basics, A complete Applet Skeleton, Applet Initialization and Termination, A key Aspect of an Applet Architecture, Requesting Repainting, using the status window, Passing parameters to Applets.<br>**Event Handling**- Two Event Handling Mechanisms.<br>**The Delegation Event Model- Events:** Event Sources, Event Listeners.<br>**Event Classes:** The ActionEvent Class, The Adjustment Event Class, The ComponentEvent Class, The Container Event Class,TheFocus EventClass,TheInputEvent Class, The ItemEvent Class, The KeyEvent Class, The MouseEvent Class, The MouseWheelEvent Class, The TextEventClass,TheWindowEvent Class.Using the Delegation Event Model-Handling Mouse Events, Handling Keyboard Events, Adapter Classes, Inner Classes, Anonymous Inner Classes | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books:**


1. Java Fundamental: A comprehensive Introduction by Herbert schildt, Dale Skrien. Tata McGraw Hill Edition 2013
2. Herbert Schildt, The Complete Reference, JAVA 7th /9th Edition, Tata McGraw Hill,2013.
3. Java 6 Programming Black Book, Dreamtech Press.2012


**Reference Books:**


1. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education,2004.
2. Uttam K Roy, Advanced JAVA programming, Oxford University press, 2015.


| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS6OE | CO1 | Apply the concepts of programming and implement programs using Java constructs. |
| | CO2 | Create classes and demonstrate object oriented programming concepts |
| | CO3 | Develop program using method overloading and inheritance. |
| | CO4 | Demonstrate applications using interfaces and run-time errors through exception handling. |
| | CO5 | Design GUI application program using Applets and event handling. |

| **Course Title: COMPUTER NETWORKS LAB** | | |
|---|---|---|
| Subject Code : **19CSL61** | Credit : 1 | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |

**Prerequisites:** Analog and Digital Electronics

**Course Objectives:**

1. Build local area network
2. To understand the physical topology of LAN, components using simulation tools
3. Analyze MAC layer performance & conduct packet analysis

| **List of Programs** | **Teaching Hours** |
|---|---|
| **PART - A**<br>1. Study CAT6 UTP EIA/TIA568A/B straight and cross-over cable crimp and test and/verify its connectivity<br>2. Install and configure network devices like hub,switch,and router and create a LAN and perform connectivity test.<br>3. Configure host IP,subnet mask and gateway in LAN<br>4. Study of basic Network configuration commands and utilities to debug the network issues.<br>5. Case Study of Campus Network Operation Center<br>6. Packet capture and header analysis by wire-shark (TCP,UDP,IP) USING WIRESHARK or any other tool.<br>7. Implement client server communication using sockets<br>PART - B<br>Following simulation experiment shall be conducted using qualnet simulator/CISCO packet tracer<br>1. Simulate a three nodes point - to – point network with duplex links between them. Set the queue size and vary the bandwidth and find the number of packets dropped.<br>2. Simulate a four node point-to-point network with the links connected as follows: N0-n2,n1-n2 and n2-n3. Apply TCP agent between n0-n3 and UDP between n1-n<br>Apply relevant applications over TC and UDP agents changing the parameter and determine the number of packets sent by TCP / UDP.<br>3. Simulate the transmission of ping messages over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.<br>4. Simulate an Ethernet LAN using n nodes (6-10), change error rate and data rate and compare throughput. | |

| | |
|---|---|
| 5. Simulate an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source/ destination.<br>6. Simulate simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.<br>7. Simulation of link state routing algorithm. | |

**Question paper pattern: For SEE , two programs from the Exercise programs list will be asked.**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL61 | CO1 | Demonstrate the use of different network cabling components and devices |
| | CO2 | Analyze performance of LAN and wireless LAN |
| | CO3 | Illustrate basic networks utilities and demonstrate client server communication. |
| | CO4 | Demonstrate working of routing algorithms. |
| | CO5 | Perform packet capture analysis the packet contents. |

| **Course Title: MACHINE LEARNING LAB** | | |
|---|---|---|
| Subject Code :**19CSL62** | **Credit : 1** | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |
| **Prerequisites: Probability & Statistics, Java/Python Programming** | | |
| **Course Objectives:**<br>• Learn implementation and applications of Machine Learning Algorithms.<br>• Understand the usage of various datasets for implementing ML Algorithms. | | |
| | | |
| **1.** Write a Program to Implement Water-Jug problem using Python.<br>**2.** Write a Program to Implement AO* Algorithm using Python<br>**3.** Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file**.**<br>**4.** For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.<br>**5.** Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.<br>**6.** Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.<br>**7.** Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.<br>**8.** Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.<br>**9.** Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API<br>**10.** Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using *k*-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.<br>**11.** Write a program to implement *k*-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem. | |

| | |
|---|---|
| 12. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs | |

**Question paper pattern: For SEE , two programs from the Exercise programs list will be asked.**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL62 | CO1 | Understand the implementation procedures for the Artificial Intelligence algorithms. |
| | CO2 | Design Python programs for various Learning algorithms. |
| | CO3 | Apply appropriate data sets to the Machine Learning algorithms. |
| | CO4 | Perform Classification and clustering of Data using ML algorithms. |
| | CO5 | Apply Machine Learning algorithms to solve real world problems. |

| **Course Title: MINI - PROJECT** | | |
|---|---|---|
| Subject Code :**19CSMP63** | **Credit : 2** | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |

**Pre-requisite:** Programming languages, Operating Systems

**Course Objectives:**

- Acquire the ability to integrate different areas of knowledge and evaluate and formulate a problem
- Acquire skills to communicate effectively and present their ideas and collaborate to work as a team.
- Understand the procedure of documentation and presentation of Mini-project

Guidelines for Mini project:

- Mini project is to be carried out individually or by a team of two to three students
- Student has to carry out literature survey to identify and formulate the problem.
- Student has to design and develop a H/W or S/W model in any domain of Computer Science.
- CIE evaluation will be done timely by a committee constituted by the department. The committee shall consist of respective guide and two faculty members.
- At the end of the semester students has to prepare and submit a project report

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CSMP63** | **CO1** | Demonstrate skills to identify and formulate given problem |
| | **CO2** | Apply basic engineering knowledge learnt in developing system individually or in group |
| | **CO3** | Evaluate current research status by conducting literature survey |
| | **CO4** | Design and develop real time application |
| | **CO5** | Apply the programming skills in software development life cycle nodel for project implementation and well-organized report |

# CURRICULUM
# FOR B.E. VII SEMESTER

## DEPARTMENT OF COMPUTER SCIENCE ANDENGINEERING

## POOJYA DODDAPPA APPA COLLEGE OFENGINEERING
**(An autonomous College under VTU)**
**KALABURAGI**

**About the institution:** The Hyderabad Karnataka Education (HKE) society founded by Late Shri. Mahadevappa Rampure, a great visionary and educationist. The HKE Society runs 46 educational institutions. Poojya Doddappa Appa College of Engineering, Gulbarga is the first institution established by the society in 1958. The college is celebrating its golden jubilee year, setting new standards in the field of education and achieving greater heights. The college was started with 50% central assistance and 50% state assistance, and a desire to impart quality technical education to this part of Karnataka State. The initial intake was 120 with degree offered in three branches of engineering viz, Civil, Mechanical and Electrical Engineering. Now, it houses 11 undergraduate courses, 10 post Graduate courses and 12 Research centers, established in Civil Engg., Electronics & Communication Engg, Industrial & Production Engg, Mechanical Engg, Electrical Engg., Ceramic Cement Tech., Information Science & Engg., Instrumentation Technology, Automobile Engg., Computer Sc. and Engg., Mathematics and Chemistry All the courses are affiliated to Visveswaraya Technological University, Belgaum. At present the total intake at UG level is 980 and PG level 193.

The college receives grant in aid funds from state government. A number of projects have been approved by MHRD /AICTE, Govt. of India for modernization of laboratories. KSCST, Govt. of Karnataka is providing financial assistance regularly for the student's projects.

The National Board of Accreditation, New Delhi, has accredited the College in the year 2005-08 for 09 UG Courses out of which 08 courses are accredited for three years and 01 course is accredited for five years. And second time accredited for Six Course in the year 2009-2012

Our college is one among the 14 colleges selected under TEQIP, sponsored by World Bank. It hasreceived a grant of Rs 10.454 Crores under this scheme for its development. The institution is selected for TEQIP phase II in year 2011 for four years. Institution is receiving a grant of Rs. 12.50 Crores under TEQIP Phase -II scheme for its development and selected for TEQIP-III as mentoring Institute for BIET Jhansi(UP).

Recognizing the excellent facilities, faculty, progressive outlook, high academic standards and record performance, the VTU Belgaum reposed abundant confidence in the capabilities of the College and the College was conferred Autonomous Status from the academic year 2007-08, to update its own programme and curriculum, to devise and conduct examinations, and to evaluate student's performance based on a system of continuous assessment. The academic programmers are designed and updated by a Board of Studies at the department level and Academic Council at the college level. These statutory bodies are constituted as per the guidelines of the VTU Belgaum. A separate examination section headed by a Controller of Examinations conducts the examinations.

At present the college has acquired the Academic autonomous status for both PG and UG courses from the academic year 2007-08 and it is one among the six colleges in the state of Karnataka to have autonomous status for both UG and PG courses.

One of the unique features of our college is, it is the first college in Karnataka State to start the Electronics and Communication Engineering branch way back in the year 1967, to join NIT Surathkal and IISc, Bangalore. Also, it is the only college in the state and one among the three colleges across the country, offering a course in Ceramic and Cement Technology. This is the outcome of understanding by faculty and management about the basic need of this region, keeping in view of the available raw material and existing Cement Industries.

Bharatiya Vidya Bhavan National Award for an Engineering College having Best Overall Performance for the year 2017 by ISTE (Indian Society for Technical Education). In the year 2000, the college was awarded as Best College of the year by KSCST, Bangalore in the state levelstudents projects exhibition.

The college campus is spread over 71 acres of land on either side of Mumbai-Chennai railway track and has a sprawling complex with gardens and greenery all around.

**About the department:** The Computer Science and Engineering department was started in the year 1984 with an intake of 40 students for UG. The department has seen phenomenal growth and now the department has increased UG intake to 120 students and offering two Post Graduation programmes: PG (Computer Science and Engineering with an intake of 25students) and PG(Computer Network and Engineering with an intake of 18 students). The department is offering research program under its recognized research center. Computer Science and Design course was started from 2021 with an intake of 60 students. The department is having state- of-the-art computing facilities with high speed internet facilities and laboratories. The department library provides useful resources like books and journals. The department has well qualified and experienced teaching faculty. The department has been conducting several faculty development programs and student training programs.

## Vision of the Institution

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

## Mission of the Institution

- To provide a high quality educational experience for students with values and ethics thatenables them to become leaders in their chosen professions.
- To explore, create and develop innovations in engineering and science through research anddevelopment activities.
- To provide beneficial service to the national and multinational industries and communitiesthrough educational, technical, and professional activities

## Vision of the Department

- To become a premier department in Computer education, research and to prepare highly competent IT professionals to serve industry and society at local and global levels.

## Mission of the Department

- To impart high quality professional education to become a leader in Computer Science and Engineering.
- To achieve excellence in Research for contributing to the development of the society.
- To inculcate professional and ethical behaviour to serve the industry.

**Program Educational Objectives (PEO)**:

| | |
|---|---|
| **PEO1:** | To prepare graduates with core competencies in mathematical and engineering fundamentals to solve and analyze computer science and engineering problems |
| **PEO2:** | To adapt to evolving technologies and tools for serving the society |
| **PEO3:** | To perform as team leader, effective communicator and socially responsible computer professional in multidisciplinary fields following ethical values |
| **PEO4:** | To encourage students to pursue higher studies, engage in research and to become entrepreneurs |

**Program Outcomes:**

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

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

**03. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

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

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

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

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

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

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

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

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

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

**Program Specific Outcomes (PSOs):**

| | |
|---|---|
| **PSO1:** | Acquire competency in hardware and software working principles to analyze and solve computing problems. |
| **PSO2:** | Design quality software to develop scientific and business applications following Software Engineering practices. |
| **PSO3:** | Apply cutting edge technologies using modern tools to find novel solutions ethically to existing problems. |

## SCHEME OF TEACHING FOR VII SEMESTER - 2023-2024

### B.E. (COMPUTER SCIENCE AND ENGINEERING)

| Code No. | Course | Hours/Week | | | | Maximum Marks | | |
|---|---|---|---|---|---|---|---|---|
| | | Lecture | Self Study | Practical | Credit | CIE | SEE | Total Mark |
| 19CS71 | Cloud Computing | 3 | 1 | 0 | 3 | 50 | 50 | 100 |
| 19CS72 | Web Application Security | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
| 19CS73X | Professional Elective- 2 | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
| 19CS74X | Professional Elective- 3 | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
| 19CS7OE | Open Elective- II | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
| 19CSL71 | Internet of Things Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
| 19CSL72 | Web Application Security Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
| 19CSS73 | Seminar | 0 | 0 | 0 | 1 | 50 | 0 | 50 |
| 19CSP74 | Project Phase – I | 0 | 0 | 2 | 2 | 50 | 50 | 100 |
| | **Total** | **16** | **1** | **6** | **21** | **450** | **400** | **850** |

**Professional Elective – 2**

| | |
|---|---|
| 19CS731 | Deep Learning |
| 19CS732 | Unix System Programming |
| 19CS733 | Social Network Analysis |
| 19CS734 | Software Project Management |

**Professional Elective – 3**

| | |
|---|---|
| 19CS741 | Wireless and Mobile Networks |
| 19CS742 | Multimedia and Virtual Reality |
| 19CS743 | Computer Graphics |
| 19CS744 | Compiler Design |

**Open Elective – II**

| | |
|---|---|
| **19CS7OE** | Web Technologies |

| **Course Title: CLOUD COMPUTING** | | | |
|---|---|---|---|
| Subject Code : **19CS71** | Credits : 3 | | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | **Self Study : 1 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | | SEE Hours: 03 |
| **Prerequisites: O**perating systems, Computer networks | | | |
| **Course objectives:**<br>• To understand Virtualization and learn Cloud Services<br>• To implement Task Scheduling algorithms.<br>• Apply Map-Reduce concept. | | | |

| **Modules** | **Teaching Hours** |
|---|---|
| **Module - I**<br>**Introduction :** Cloud Computing at a Glance, The Vision of Cloud Computing, Defining a Cloud, A Closer Look, Cloud Computing Reference Model, Characteristics and Benefits, Challenges Ahead, Historical Developments, Distributed Systems, Virtualization, Web 2.0, Service-Oriented Computing, Utility-Oriented Computing, Building Cloud Computing Environments, Application Development, Infrastructure and System Development, Computing Platforms and Technologies, Amazon Web Services (AWS), Google App Engine, Microsoft Azure, Hadoop, Force.com and Salesforce.com, Manjrasoft Aneka Virtualization, Introduction, Characteristics of Virtualized, Environments Taxonomy of Virtualization Techniques, Execution Virtualization, Other Types of Virtualization, Virtualization and Cloud<br>Computing, Pros and Cons of Virtualization, Technology. Case Study Containers, Dockers. | **9 Hrs** |
| **Module - II**<br>Cloud Computing Architecture, Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Definition, Cloud Interoperability and Standards Scalability and Fault Tolerance Security, Trust, and Privacy Organizational Aspects Aneka: Cloud Application Platform, Framework Overview, Anatomy of the Aneka Container, From the Ground Up: Platform Abstraction Layer, Fabric Services, foundation Services, Application Services, Building Aneka Clouds, Infrastructure Organization, Logical Organization, Private Cloud Deployment Mode, Public<br>Cloud Deployment Mode, Hybrid Cloud Deployment Mode, Cloud Programming and Management, Aneka SDK, Management Tools. | **8 Hrs** |

| | |
|---|---|
| **Module - III**<br>**Concurrent Computing:** Thread Programming, Introducing Parallelism for Single Machine Computation, Programming Applications with Threads, What is a Thread, Thread APIs, Techniques for Parallel Computation with Threads, Multithreading with Aneka, Introducing the Thread Programming Model, Aneka Thread vs. Common Threads, Programming Applications with Aneka Threads, Aneka Threads Application Model, Domain<br>**Decomposition:** Matrix Multiplication, Functional Decomposition:Sine, Cosine, and Tangent. High-Throughput Computing: Task Programming, Task Computing, Characterizing Task, Computing Categories, Frameworks forTask Computing, Task-based Application Models, Embarrassingly Parallel Applications, Parameter Sweep Applications, MPI Applications, Workflow Applications with Task Dependencies, Aneka Task-Based Programming, Task Programming Model, Developing Applicationswith the Task Model, developing Parameter Sweep Application, Managing Workflows. | **09Hrs** |
| **Module - IV**<br>Data Intensive Computing: Map-Reduce Programming, What is Data-Intensive Computing, Characterizing Data-Intensive Computations, Challenges Ahead, Historical Perspective, Technologies for Data-Intensive Computing, Storage Systems, Programming Platforms, Aneka Map Reduce Programming, Introducing the Map Reduce Programming Model, Example Application. | **08 Hrs** |
| **Module - V**<br>Cloud Platforms in Industry, Amazon Web Services, Compute Services, Storage Services, Communication Services, Additional Services, Google App Engine, Architecture and Core Concepts, Application Life-Cycle, Cost Model, Observations, Microsoft Azure, Azure Core Concepts, SQL Azure, Windows Azure Platform Appliance. Cloud Applications Scientific Applications, Healthcare: ECG Analysis in the Cloud, Social Networking, Media Applications, Multiplayer Online Gaming. | **08 Hrs** |
| **Question paper pattern:**<br>The question paper will have ten questions.<br>There will be 2 questions from each module, covering all the topics from a module.<br>The students will have to answer 5 full questions, selecting one full question from each module**.** | |
| **Text Book:**<br>1. International Edition - Rajkumar Buyya, Christian Vecchiola, and Thamarai selvi, Mastering Cloud Computing, Morgan Kaufmann, ISBN: 978-0-12-411454-8, Burlington, Massachusetts,USA, May 2013. | |

**REFERENCE BOOKS**

1. Paul Goransson and Chuck Black, Software Defined Networks: A Comprehensive Approach, 1st edition, 2014, Morgan Kaufmann Publishers, Inc., San Francisco. ISBN-13: 978- 0124166752, ISBN-10:012416675

2. T. Erl, R. Puttini, and Z. Mahmood, Cloud Computing: Concepts, Technology & Architecture ISBN-10: 0133387526 • ISBN-13: 9780133387520 ©2013 • Prentice Hall .

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS71 | CO1 | Describe Cloud Computing setup and applications using different architecture and understand concept of Virtualization . |
| | CO2 | Demonstrate various cloud reference models and deployment modes |
| | CO3 | Develop and deploy cloud application using popular cloud platforms. |
| | CO4 | Understand Data intensive computing and apply Map Reduce |
| | CO5 | Describe the importance of cloud computing driven commercial systems. |

| **Course Title: WEB APPLICATION SECURITY** | | |
|---|---|---|
| Subject Code : **19CS72** | Credit : **04** | CIE: 50 |
| Number of Lecture Hours/Week | **04 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 52 | SEE Hours: 03 |
| Prerequisites: Computer Network | | |

**Course Objectives:**

- Gain understating of threat surface.
- To discover security flaws in web applications.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Web Application Insecurity And Defense Mechanism:** The Evolution of Web Applications, Web Application Security, Key Problem Factors, Handling User Access, Handling User Input, Handling Attackers<br>**Web application technologies:** HTTP Protocol, Web Functionality, Encoding Schemes | **10 hrs** |
| **Module II**<br>**Mapping Application:** Enumerating Content and functionality, Analyzing application. **Bypassing Client-side controls:** Transmitting Data via Client Capturing User Data: HTML FORMS, Browser Extensions<br>**Attacking Authentication:** Authentication technologies, Design flaws in authentication, Implementation flaws in authentication, Securing authentication | **11 hrs** |
| **Module III**<br>**Attacking Session Management:** The Need for state, Weaknesses in token generation, Weaknesses in session token handling, Securing session management. **Attacking Access Controls:** Common vulnerabilities, Attacking access controls, Securing access controls. | **10 hrs** |
| **Module IV**<br>**Attacking Data Stores:** Injecting into interpreted contexts, Injecting into SQL, Injecting into NoSQL, injecting into XPath, Injecting into LDAP. **Attacking Back-end components:** Injecting OS Commands, Manipulating File Paths, Injecting into XML Interpreters, Injecting into Back-end HTTP Requests, Injecting into Mail Services. | **11 hrs** |

| Module V | |
|---|---|
| **Attacking Application Logic:** The Nature of Logic Flaws , Real – World Logic Flaws, Ex.1 Fooling a password change function , Ex.2 Breaking the bank, Ex.3 Cheating on bulk discounts, Ex.4 Invalidating input validation, Ex.5Racing against the login, avoiding logic flaws.<br>**Attacking Users: Cross-Site Scripting**: Varieties of XSS,XSS Attacks in Action, Finding and Exploiting XSS vulnerabilities, Preventing XSS Attacks. | **10 hrs** |

**TEXTBOOK:**

1. Web Application Hacker‟s Handbook, Dafydd Stutarf, Marcus Pinto, Wiley, 2nd Edition, 2011

**REFERENCE BOOKS:**

1. Hacking Exposed Web Applications, Third Edition, 3rd Edition, by Joel Scambray, Vincent Liu, Caleb Sima. Released October 2010. Publisher(s): McGraw-Hill.
2. Hacking: The Art of Exploitation by Jon Erickson, 2nd Edition, Feb 2008
3. Penetration Testing: A Hands-On Introduction to Hacking Paperback by Georgia Weidman, June 2014

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS72 | CO1 | Describe vulnerabilities associated with web applications. |
| | CO2 | Analyze the application and identify authentication design flaws |
| | CO3 | Evaluate session management and access control vulnerabilities and adopt security methods. |
| | CO4 | Demonstrate SQL and OS injection in an ethical way. |
| | CO5 | Illustrate nature of logic flaws in real world applications. |

| Course Title: **DEEP LEARNING** | | |
|---|---|---|
| Subject Code : **19CS731** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** Machine learning | | |
| **Course Objectives:**<br>• Understand complexity of Deep Learning algorithms and their limitations.<br>• Be capable of performing experiments in Deep Learning using real-world data. | | |

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Introduction to Deep Learning:** Introduction to deep learning , Biological & artificial neurons ANN & its layer, Exploring activation functions, Forward propagation in ANN, How does ANN learn ,Debugging gradient descent with gradient checking. Getting to Know TensorFlow: Introduction to TensorFlow, Understanding computational graphs and sessions, Variables, constants and placeholders, Introducing TensorBoard , Handwritten digit classification using TensorFlow , Introducing eager execution, Math operations in TensorFlow, TensorFlow 2.0 and Keras, Keras or TensorFlow. | **09Hrs** |
| **Module II**<br>**Introduction to RNN:** Generating Song Lyrics Using RNN, Introducing RNNs Generating song lyrics using RNNs, Different types of RNN architectures.<br>**Improvements to the RNN:** Improvements to the RNN, LSTM to the rescue, Gated recurrent units, Bidirectional RNN ,Going deep with deep RNN, Language translation using the seq2seq model. | **09 Hrs** |
| **Module III**<br>**Demystifying Convolutional Networks:** Demystifying Convolutional Networks, Introduction to CNNs ,The architecture of CNNs ,The math behind CNNs , Implementing a CNN in TensorFlow, CNN architectures, Capsule networks, Building Capsule networks in TensorFlow.<br>Case study | **08 Hrs** |

| | |
|---|---|
| **Module IV**<br>L**earning Text Representations:** Learning Text Representations ,Understanding the word2vec model ,Building the word2vec model using gensim, Visualizing word embeddings in TensorBoard, Doc2vec Understanding ,skip-thoughts algorithm ,Quick-thoughts for sentence embeddings. | **08Hrs** |
| **Module V**<br>**Generating Images Using GANs:** Generating Images Using GANs,<br>Differences between discriminative and generative models.<br>DCGAN – Adding convolution to a GAN, Deconvolution generator, convolutional discriminator.<br>**Learning More about GANs:** Conditional GAN, Loss Function of CGAN, Generating specific digits using CGAN, Understanding InfoGAN, Exploring Mutual Information, Architecture of InfoGAN, Translating images using CycleGAN, Role of generators, Role of discriminators, Loss Function, Cycle Consistency Loss, Stack GAN, Architecture of StackGANs. Introduction to auto encoder. | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**

1.Sudharsan Ravichandiran, "Hands on deep learning algorithms with python", Packt Publishing , July 2019, ISBN: 9781789344158.

**REFERENCE BOOKS:**

1.Goodfellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016
2. Francois Chollet, Deep Learning with Python, Manning Publications, 2018.

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

| Course Code | CO # | Course Outcome (CO |
|---|---|---|
| 19CS731 | CO1 | Understand the concepts of Deep Learning, TensorFlow, its main functions, operations and the execution pipeline. |
| | CO2 | Understand Recurrent Neural Networks(RNN), Implement different architectures of RNN in Tensorflow. |
| | CO3 | Learn convolutional neural networks, Implement different architectures of CNN in Tensorflow. |
| | CO4 | Demonstrate Text Representations and Build the word2vec model using gensim and interpret the results. |
| | CO5 | Build different architectures of GANS in Tensorflow. |

| Course Title: **UNIX SYSTEM PROGRAMMING** | | |
|---|---|---|
| Subject Code : **19CS732** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** Unix Shell Programming, operating systems. | | |
| **Course Objectives:**<br>• To understand fundamental design of Unix operating system.<br>• To use Unix system calls to build an application service over Unix operating system.<br>• Gain knowledge of Unix internals. | | |

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module –I**<br>**Introduction:** UNIX and ANSI Standards, X/Open Standards, POSIX APIs, POSIX development Environment, API Common Characteristics, File types, Attributes, Inodes in UNIX System V, APIs to Files, UNIX Kernel support for files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links. | **09 Hrs** |
| **Module-II**<br>**UNIX File APIs :** General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO file APIs, Symbolic Kink File APIs, General File Class, regfile Class for Regular Files, dirfile Class for Directory File, FIFO file Class, Device File Class, Symbolic Link File Class, File Listing Program. | **09 Hrs** |
| **Module-III**<br>**UNIX Processes :** The Environment of a UNIX Process, main function, Process termination, command-line arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp functions, getrlimit, setrlimit functions, UNIX Kernel Support for Processes. | **08 Hrs** |
| **Module- IV**<br>**Process Control and Signals :** Process Identifiers, fork, vfork, exit, wait, waitpid, race conditions, exec functions, changing user ids, Interpreter files, systems function, Process Accounting, User Identification, Process Times, Signals : The Unix Kernel Support for Signals, Signal Mask, sigaction, the SIGCHLD signal and waitpid function, the sigsetjmp and siglongjmp functions, Kill, Alarm, Interval Timers. | **08 hrs** |
| **Module – V**<br>**Daemon Processes and Inter Process Communication :** Daemon Processes, Daemon Characteristics, Daemon Conventions, client-servier Model. Inter Process Communication, Pipes, popen, pclose, FIFOs, Message Queues, Semaphores. | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books**

1. W. Richard Stevens, "Advanced Programming in the UNIX Environment",3rd Edition, Pearson Eductation/PHI, 2013.
2. Terrance Chan, "Unix System Programming Using C++", Prentice Hall India, 1999.

**Reference Books**

1. Marc J Rochkind, "Advanced Unix Programming", 2nd Edition, Pearson Education, 2005.
2. Maurice J Bach, "The design of the UNIC Operating System", Pearson Education/PHI, 1987.
3. D A Patterson and J L Hennessy, "Computer Architecture: A Quantitative Approach", Harcourt Asia, Morgan Kaufmann, 1999.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS732 | **CO1** | Discuss UNIX , ANSI open Standard and understand kernel support for FILES. |
| | **CO2** | Illustrate different UNIX file API |
| | **CO3** | Discuss UNIX process and kernel support for processes |
| | **CO4** | Demonstrate process control API and signal handling |
| | **CO5** | Describe Daemon processes and IPC system calls. |

| **Course Title: SOCIAL NETWORK ANALYSIS** | | |
|---|---|---|
| Subject Code : **19CS733** | Credit : 03 | CIE: 50 |
| Number of Lecture Hours/Week | **03 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** Machine Learning, Mathematical Foundations of Computer Science | | |

**Course Objectives:**

- To understand the concept of semantic web and related applications,
- To learn knowledge representation using ontology,
- To understand human behaviour in social web and related communities,
- To learn visualization of social networks.

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Introduction:** Introduction to Semantic Web: Limitations of current Web – Development of Semantic Web – Emergence of the Social Web – Social Network analysis: Development of Social Network Analysis – Key concepts and measures in network analysis – Electronic sources for network analysis: Electronic discussion networks, Blogs and online communities – Web-based networks – Applications of Social Network Analysis. | **10 Hrs** |
| **Module II**<br>**Modelling, Aggregating and Knowledge, Representation:**<br>Ontology and their role in the Semantic Web: Ontology-based knowledge Representation – Ontology languages for the Semantic Web: Resource Description Framework – Web Ontology Language – Modelling and aggregating social network data: State-of-the-art in network data representation<br>– Ontological representation of social individuals – Ontological representation of social relationships – Aggregating and reasoning with social network data – Advanced representations | **08 Hrs** |
| **Module III**<br>**Extraction and mining communities in web social networks**<br>Extraction evolution of Web Community from a Series of Web Archive-Detecting communities in social networks-Definition of Community-Evaluating Communities-Methods for community detection and mining-Application of community mining algorithms-Tools for detecting communities social network infrastructures and communities-Decentralized online social networks-Multi-Relational characterization of dynamic social network communities. | **08 Hrs** |

| | |
|---|---|
| **Module IV**<br><br>**Predicting Human Behaviour and Privacy Issues :** Understanding and predicting human behaviour for social communities – User data management – Inference and Distribution – Enabling new human experiences – Reality mining – Context – Awareness – Privacy in online social networks – Trust in online environment – Trust models based on subjective logic – Trust network analysis – Trust transitivity analysis – Combining trust and reputation – Trust derivation based on trust comparisons – Attack spectrumand countermeasures. | **08 Hrs** |
| **Module V**<br>**Visualization and Applications of Social Networks:** Graph theory – Centrality – Clustering – Node-Edge Diagrams – Matrix representation – Visualizing online social networks, Visualizing social networks with matrix-based representations – Matrix and Node-Link Diagrams<br>– Hybrid representations – Applications – Cover networks – Community welfare – Collaboration networks – Co-Citation networks. | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books:**

1. Peter Mika, ―Social Networks and the Semantic Web, First Edition, Springer 2007.
2. Borko Furht, ―Handbook of Social Network Technologies and Applications, 1st Edition, Springer, 2010

**Reference Books:**

1. Hanneman , Robert A. and Mark Riddle. 2005. Introduction to social network methods.

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS733 | CO1 | Develop semantic web related applications. |
| | CO2 | Understand Representation knowledge using ontology. |
| | CO3 | Demonstrate human behaviour Prediction in social web and related communities. |
| | CO4 | Understand Visualization of social networks. |
| | CO5 | Illustrate Mining communities in social networks |

| **Course Title: SOFTWARE PROJECT MANAGEMENT** | | |
|---|---|---|
| Subject Code : **19CS734** | Credit : 03 | CIE: 50 |
| Number of Lecture Hours/Week | 03 | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites: Software Engineering** | | |

**Course Objectives**:

- Understand the fundamental principles of project management
- Be familiar with different methods and techniques used for Project management.
- Exposure to issues and challenges faced while doing s/w project management.
- Able to perform Project Scheduling ,tracking, Risk Analysis, Quality management and Project cost estimation

| **MODULES** | **Teaching Hours** |
|---|---|
| **Module I**<br><br>**Software Management & Economics SDLC**: Waterfall model Conventional Software Management Performance Evolution of Software Economics – Software economics Pragmatic software cost estimation Reducing software product size Improving software processes Improving team effectiveness Improving automation through software environment. | **09 Hrs** |
| **Module II**<br><br>**The Old And The New Way Of Project Management: The** principles of conventional software engineering Principles of modern software management, Transitioning to an iterative process Basics of Software estimation – Effort and Cost estimation techniques COSMIC Full function points COCOMO-I COCOMO II A Parametric Productivity Model - Staffing Pattern. | **08 Hrs** |
| **Module III**<br><br>**Software Management Process Framework**: Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases. Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts Model based software architectures: A Management perspective. Model based software architectures: Technical perspective Work Flows of the process: Software process workflows Iteration workflows Checkpoints of the process: Major milestones, Minor Milestones, Periodic status assessment. | **09 Hrs** |
| **Module IV**<br><br>**Project Organization and Planning**: Work breakdown structures Planning guidelines. The cost and schedule estimating process The iteration planning process Pragmatic planning Line-of-Business organizations Project organizations, Evolution of organizations Process automation - Automation building Blocks The project environment. | **08 Hrs** |

| **Module V** | **08 Hrs** |
|---|---|
| **Project Control and Process Instrumentation:** The Seven-Core metrics: Management indicators The Seven-Core metrics: Quality indicators Life-Cycle expectations, Pragmatic software metrics, Metrics automation Modern project profiles Next generation software economics Modern process transitions. | |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books**

1. Walker Royce, "Software Project Management", 1st Edition, Pearson Education, 2006.

**References Books**

1 Bob Huges, Mike Cotterell, Rajib Mall, Software Project Managemen, 6 th Edition, Tata McGraw Hill, 2017.
2. SA Kelkar, Software Project Management: A Concise Study, 3 rd Edition, PHI, 2013.
3. Joel Henry, Software Project Management: A Real-World Guide to Success, Pearson Education, 2009.
4. Pankaj Jalote, Software Project Management in Practice, Pearson Education, 2015.
5. https://ocw.mit.edu/courses/engineering-systems-division/esd-36-system-projectmanagement-fall-2012/
6. https://uit.stanford.edu/pmo/pm-life-cycle

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS734 | CO1 | Identify the different project contexts and suggest an appropriate management strategy. |
| | CO2 | Practice the role of professional ethics in successful software development. |
| | CO3 | Identify and describe the key phases of project management. |
| | CO4 | Determine an appropriate project management in organizing and planning . |
| | CO5 | Analyze the concepts Project control and Process instrumentation |

| **Course Title: WIRELESS AND MOBILE NETWORKS** | | |
|---|---|---|
| Subject Code: **19CS741** | **Credits: 3** | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |

**Prerequisites: Computer Networks**

**Course Objectives:**

- To learn the basics of Wireless voice and data communications technologies.
- To study the working principles of wireless LAN and its standards.
- To build knowledge on various Mobile Computing algorithms.
- To build skills in working with Wireless application Protocols to develop mobile content applications

| **Modules** | **Teaching Hours** |
|---|---|
| **Module - I**<br>**Wireless Communication Fundamentals :** Introduction – Wireless transmission – Frequencies for radio transmission – Signals – Antennas Signal Propagation – Multiplexing – Modulations – Spreadspectrum – MAC – SDMA – FDMA – TDMA – CDMA – Cellular Wireless Networks. | **09 Hrs** |
| **Module – II**<br>**Telecommunication Networks :** Telecommunication systems – GSM – GPRS – DECT – UMTS – IMT-2000 –Satellite Networks -Basics – Parameters and Configurations – Capacity Allocation – FAMA and DAMA – Broad cast Systems – DAB - DVB. | **09 Hrs** |
| **Module –III**<br>**Wireless LAN :** Wireless LAN – IEEE 802.11 - Architecture – services – MAC – Physical layer – IEEE 802.11a 802.11b standards – HIPERLAN – Blue Tooth. | **08 Hrs** |
| **Module –IV**<br>**Mobile Network Layer :** Mobile IP – Dynamic Host Configuration Protocol - Routing – DSDV – DSR –Alternative Metrics | **08 Hrs** |
| **Module – V**<br>**Transport And Application Layers :** Traditional TCP – Classical TCP improvements – WAP, WAP 2.0 | **08 Hrs** |

**Question paper pattern:**

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

**Text Books :**

1. Jochen Schiller, "Mobile Communications", PHI/Pearson Education, Second Edition, 2004.

**Reference Books:**

1. Kaveh Pahlavan, Prasanth Krishnamoorthy, "Principles of Wireless Networks", PHI/Pearson Education, 2003.

2. Uwe Hansmann, Lothar Merk, Martin S. Nicklons and Thomas Stober, "Principles of Mobile Computing", Springer, New York, 2003.
3. Hazysztof Wesolowshi, "Mobile Communication Systems", John Wiley and Sons Ltd, 2002.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS741 | CO1 | Understand the concept of Wireless Communication Fundamentals. |
| | CO2 | Demonstrate the concepts of wireless technologies. |
| | CO3 | Illustrate Wireless Architecture and services. |
| | CO4 | Demonstrate routing protocols . |
| | CO5 | Describe Transmission control Protocol and Wireless Application Protocol |

| **Course Title: MULTIMEDIA AND VIRTUAL REALITY** | | |
|---|---|---|
| Subject Code: **19CS742** | **Credits : 3** | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** DBMS, NETWORKING | | |

**Course Objectives:**

- To learn the basics of Multimedia Technology.
- To Learn Compression Techniques and Methodology
- To Learn about Applications of Multimedia design and Information Management
- Illustrate process of creating virtual environments

| **Modules** | **Teaching Hours** |
|---|---|
| **Module - I**<br>**INTRODUCTION :** Multimedia Elements, Multimedia Applications, Multimedia System Architecture, Evolving Technologies for Multimedia systems, Defining Objects for Multimedia Systems, Multimedia Data Interface Standards, The Need for Data Compression, Multimedia Databases | **09 Hrs** |
| **Module – II**<br>**COMPRESSION TECHNIQUES**: Basic concepts of Compression. Binary Image Compression: JPEG Compression. Features of JPEG2000. Video Compression: MPEG- 1&2 Compression Schemes, MPEG-4 Natural Video Compression. Audio Compression: Introduction to speech and Audio Compression, MP3 Compression Scheme. Compression. Of synthetic. | **09 Hrs** |
| **Module –III**<br>**MULTIMEDIA APPLICATION DESIGN** : Multimedia Application classes,Types of Multimedia Systems, Virtual reality design, Components of Multimedia System, Organizing Multimedia Database, Applications of Workflow design issues, Distributed Application design Issues. | **08 Hrs** |
| **Module -IV**<br>**INTRODUCTION TO VIRTUAL REALITY:** What is Virtual Reality, Modern VRExperiences, Hardware, Software, Human Physiology and Perception<br><br>**VISUAL PERCEPTION IN VIRTUAL REALITY:** Perception of Depth, Perception of Motion, Perception of Color, Combining Sources of Information | **08 Hrs** |
| **Module - V**<br><br>**AUDIO IN VIRTUAL REALITY**: The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering<br><br>**EVALUATING VR SYSTEMS AND EXPERIENCES:** Perceptual Training, Recommendations for Developers, Comfort and VR Sickness, Experiments on Human Subjects | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books :**

1. Multimedia System Design, Andleigh and Thakarar , PHI, 2015
2. Virtual Reality Systems, R . A . Earnshaw, M.A. Gigante, H . Jones, ACADEMIC PRESS, 2018.

**Reference Books :**

1. Multimedia Computing Communication and Application, Steinmetz, Pearson Edn.Virtual Reality Systems , John Vince, Pearsn Education.
2. Fundamentals of Computer Graphics and Multimedia, D.P. Mukherjee, PHI

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS742 | **CO1** | Demonstrate the concepts of Multimedia Computing |
| | **CO2** | Analyse the working of Compression techniques |
| | **CO3** | Explore the Multimedia Application classes and Design issues |
| | **CO4** | Demonstrate the Visual perceptions of Virtual Reality |
| | **CO5** | Understand Virtual Reality design Concepts and Evaluation |

| Course Title: **COMPUTER GRAPHICS** | | |
|---|---|---|
| Subject Code :**19CS743** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** Mathematics, C/C++ . | | |
| **Course Objectives:**<br>• Identify and explain the core concepts of computer graphics.<br>• Apply graphics programming techniques to design, and create computer graphics scenes.<br>• Create effective OpenGL programs to solve graphics programming issues, including 3D transformation, objects modeling, color modeling, lighting, textures. | | |
| **MODULES** | | **Teaching Hours** |
| **Module –I**<br>**Introduction**: Applications of computer graphics, A graphics system, Images: Physical and synthetic, The human visual system, The pinhole camera, The synthetic camera Model, The programmer„s interface, Graphics architectures, Input devices.<br>**Graphics programming:** The Sierpinski gasket, The OpenGL API, Primitives and attributes, Color, Viewing, Control functions, The Gasket program, Polygons and recursion, The three-dimensional gasket. | | **08 Hrs** |
| **Module-II**<br>**Raster Graphics Algorithms:** Overview, Scan converting lines, Scan converting circles, Filling rectangles, Filling polygons, Filling Ellipse arcs, Thick primitives, Clipping in a raster world, Clipping lines, Clipping circles and ellipses, Clipping polygons, Generating characters, Antialiasing.<br>**Input and Interaction:** Interaction, Input devices, Clients and servers, Display lists, Display lists and modeling, Programming event-driven input, Menus, Picking, Building interactive models, Animating interactive programs, Design of interactive programs. | | **09 Hrs** |
| **Module-III**<br>**Geometric Objects and Transformations :**Scalars, points, and vectors, Three-dimensional primitives, Coordinate systems and frames, Modeling a colored cube, Affine transformations, Rotation, translation and scaling, Transformation in homogeneous coordinates, Concatenation of transformations, OpenGL transformation matrices, Interfaces to three-dimensional applications, Quaternions. | | **08 Hrs** |

| | |
|---|---|
| **Module- IV**<br>**Viewing:** Classical and computer viewing, Positioning of the camera, Simple projections, Projections in OpenGL, Hidden-surface removal, Walking through a scene, Parallel projection matrices, Perspective-projection matrices, **Lighting and Shading :** Light and matter, Light sources, The Phong reflection model, Computation subdivisions, Light surfaces in OpenGL, Specification of materials in OpenGL | **09 Hrs** |
| **Module – V**<br>**Visible Surface Detection and Animation:** Visible Surface Detection: Classification of Visible Surface Detection algorithm, Back Surface detection method, Depth Buffer method, Area Subdivision method<br>**Animation:** Introduction to Animation, Traditional Animation Techniques, Principles of Animation, Key framing: Character and Facial Animation, Deformation, Motion capture | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text books:**

1. Edward Angel, *"Interactive Computer Graphics A Top-Down Approach with OpenGL",* 2nd Edition, Addison-Wesley,2000.
2. Foley, Van Dam, Feiner, Hughes, *"Computer Graphics: Principles and Practice",* Addison Wesley, ISBN0-201-12110-7.,1997
3. Samit Bhattacharya, "Computer Graphics", Oxford Publication

**Reference Books:**

1. D. Hearn, M.P. Baker, *"Computer Graphics ",* 3rd Edition, Prentice Hall,2004.
2. Interactive Computer Graphics A Top-Down Approach with OpenGL -Edward Angel, 5th Edition, Addison-Wesley,2008
3. Computer Graphics Using OpenGL – F.S. Hill,Jr. 2nd Edition, Pearson Education, 2001.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS743 | CO1 | Describe the basics of Computer Graphics, Graphical inputdevices and graphics programming |
| | CO2 | Implement Raster Graphics Algorithms for primitive operation clipping, filling using interactive programs. |
| | CO3 | Investigate three dimensional transformations and itsinterfaces using OpenGL |
| | CO4 | Explain types of projection, rendering, lighting and shading |
| | CO5 | Analyze Visualization techniques and animation. |

| **Course Title: COMPILER DESIGN** | | |
|---|---|---|
| Subject Code: **19CS744** | **Credits : 3** | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** Finite Automata and Formal Language. | | |
| **Course Objectives:**<br>• Understand the stages of compiler.<br>• Understand syntax analysis, various types of parsers, intermediate code generation and code generation. | | |

| **Modules** | **Teaching Hours** |
|---|---|
| **Module I**<br>**Introduction to Compiling:** Compilers, Analysis of the source program, Phases of the compiler, cousins of compilers, grouping of phases, compiler construction tools.<br>**A Simple one pass compiler:** Syntax definition, Syntax-directed Translation, Parsing, A translator for simple expressions, Lexical analysis, Incorporating a symbol table, abstract stack machines, Putting the techniques together.<br>**Lexical Analysis:** The role of the lexical analyzer, Input buffering, Specifications of tokens, Recognition of tokens, A language for specifying lexical analyzer. | **09 Hrs** |
| **Module II**<br>**Syntax Analysis:** The role of parser, context free grammars, Writing a grammar, Top-down Parsing, Bottom-Up parsing, Operator precedence parsing, LR-parser using ambiguous grammars, parser generators. | **09 Hrs** |
| **Module III**<br>**Syntax-Directed Translations:** Syntax directed definitions, construction of syntax trees, bottom-up evaluation of S-attributed definitions, L-attributed definitions, top-down translations, bottom-up evaluation of inherited attributes, Recursive evaluators, space for attribute values at compile time, assigning space for compiler construction time, Analysis of syntax directed definition. | **08 Hrs** |
| **Module IV**<br>**Type Checking:** Type systems, specification of simple type checker, equivalence of type expressions, polymorphic functions, algorithm for unification.<br>**Runtime Environments:** Source language issues, storage organization, storage allocation strategies, access to non local names, parameter passing.<br>**Intermediate Code Generation**: Intermediate languages, declarations, Assignment statements, Boolean expressions case statements, back patching, procedure calls. | **08 Hrs** |

| **Module V** | |
|---|---|
| **Code Generation:** Issues in the design of code generator, target machine, runtime storage management, Basic blocks and flow graphs, next-use information, simple code generator, register allocation and assignment, the DAG representation of basic blocks, peephole optimization, generating code from DAG. | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text book:**

1. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman: Compilers - Principles,Techniques and Tools, 2nd Edition, Pearson, 2011.

**Reference Books:**

1. Andrew W Apple: Modern Compiler Implementation in C, Cambridge University Press,1997
2. Charles N. Fischer, Richard J. leBlanc, Jr.: Crafting a Compiler with C, Pearson, 1991.
3. Kenneth C Louden: Compiler Construction Principles & Practice, Cengage Learning,2012

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS744** | **CO1** | Understand phases of compilers and implementation of lexical analyzer. |
| | **CO2** | Design different parsers using Context free grammars. |
| | **CO3** | Demonstrate syntax directed definitions, construction of syntax trees, and analysis of syntax directed definitions. |
| | **CO4** | Describe specification of type checker, storage allocation strategies and techniques for intermediate code generation. |
| | **CO5** | Describe techniques for machine code generation. |

| Course Title: **WEB TECHNOLOGIES** | | |
|---|---|---|
| Subject Code : **19CS7OE** | Credits : 3 | CIE: 50 |
| Number of Lecture Hours/Week | 3 (Theory) | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |

**Pre-requisite:**
Basics of any Programming Language

**Course objectives:**

- Provide the principles and practical programming skills of developing Webapplications.
- Enables students to develop skills for creating dynamic webpages using JavaScripts, XML, PHP as Server side Scripting.

| Modules | Teaching Hours |
|---|---|
| **Module-I**<br>**Fundamentals of Web, XHTML-1:** Internet, WWW, Web Browsers, and Web servers; URLs; MIME; HTTP, Security; The Web Programmers Toolbox, XHTML; Origins and Evolution of HTML and XHTML; Basic Syntax; Standard XHTML document Structure; Basic text Markup. XHTML2: Images; Hypertext Links; Lists; Tables; Forms; Frames; Syntactic Differences between HTML.<br>**CSS:** Introduction ; Levels of Style Sheets; Style Specification formats; Selector Forms; Property value forms; Font properties; ListProperties; Color; Alignment of Text; The Box Model; Background Images; The  and  tags; Conflict Resolution. | **08 Hrs** |
| **Module-II**<br>**JavaScript:** Overview of JavaScript; Object Orientation and JavaScript; General syntactic characteristics; Primitives, operations, and Expressions; Screen output and keyboard input; Control statements; Object creation and modification Arrays; Functions; Constructor, Pattern Matching using regular expression; Errors in Scripts; Examples.<br>**JavaScript and HTML Documents**: The JavaScript Execution Environment; The Document Object Model; Element Access in JavaScript; Events and event handling; Handling Events from the Body Elements, Button Elements, Text box and Password elements; The DOM 2 event model; The Navigator object; DOM 2 event model; the navigator object;<br>DOM tree traversal and modification. | **09 Hrs** |
| **Module-III**<br>**Dynamic Documents With Java script:** Introduction to dynamic documents; Positioning elements; Moving elements; Element visibility; Changing colors and fonts; Dynamic content; Stacking elements; Locating the mouse cursor, reaching to mouse click; Slow Movement of elements; Dragging and dropping elements. | **08 Hrs** |

| | |
|---|---|
| **Module-IV**<br>**XML:** Introduction; Syntax; Document structure, Document Type definitions; Namespaces ; XML schemas ; Displaying raw XML documents ; Displaying XML documents with CSS ; XSLT style sheets ; XML Processors; Web services. | **09 Hrs** |
| **Module-V**<br>PHP: Origins and uses of PHP; Overview of PHP; General Syntactic Characteristics; Primitive; Operations and Expressions; Output; Control Statements; Arrays; Functions; Pattern Matching; Form Handling, Files, Cookies; Session Tracking. | **08 Hrs** |

**Question paper pattern:**

The question paper will have ten questions.

There will be 2 questions from each module, covering all the topics from a module.

The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text books:**

1. Robert W. Sebsta, *"Programming the World Wide Web*"- 6th Edition, Pearson Education, 2011.
2. Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", Pearson Education India, 1st Edition, 2016
3. Jeffrey C. Jackson, "Web Technologies--A Computer Science Perspective", Pearson Education, 1st Edition, 2006.

**Reference Books:**

1. M Deitel, P.J. Deitel, A.B Goldberg, ***"Internet & World Wide Web How to H Program"***- 3rd Edition, Pearson Education/PHI, 2004
2. Chris Bates, *"**Web Programming Building Internet Applications*"**- 3rd Edition, Wiley India, 2006.
3. Xue Bai Et al, Thomson, *"**The Web Warrior Guide to Web Programming"***- 2003.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS7OE** | **CO1** | Apply the knowledge of HTML tags and CSS to design web pages. |
| | **CO2** | Create dynamic web application using Java script and Document object model |
| | **CO3** | Create dynamic documents using Java Scripting, |
| | **CO4** | Create XML documents with CSS, XSLT and Illustrate use of XML processors, web services. |
| | **CO5** | Create PHP documents for server side scripting |

| Course Title: **INTERNET OF THINGS LAB** | | |
|---|---|---|
| Subject Code : **19CSL71** | Credits :1 | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |

**Prerequisite:** Analog & Digital Electronics

**Course Objectives :**

- Upgrading a simple thing to a smart thing by adopting IoT technology.
- Design & develop IOT smart systems with sensors which can acquire data through sensors.
- Inter-connecting smart devices for acquiring data for analysis.
- Explore the concept of a smart city.
- Developing control systems used in today's smart world

| **List of Programs** | **Teaching Hours** |
|---|---|
| 1. Familiarization with Arduino & Raspberry Pi.<br>Develop a working model fora)<br>LED ON for 2 min &<br>b) Sound the buzzer<br>c) LED OFF for 2mins<br>d) Sound the buzzer<br>e) Repeat a) to d) | |
| 2. Develop a working model for counting number of Spokes in a wheel using IR/LDR Sensor. | |
| 3. Develop a working model for knowing Temperature and Humidity in air using DHT11 Sensor. | |
| 4. Develop a working model for measuring moisture in soil show Red led ON if moisture level is less than a threshold and if moisture level is more than threshold level show Green LED ON. | |
| 5. Develop a working model display numbers from ∞ to 99m Seven segment display. | |
| 6. Develop a working model to display four different messages on LCD display button press / with time interval. | |
| 7. Develop a working model to switch on 3 different Bulbs based on 3 different keys using Relays. | |

| | |
|---|---|
| 8. Develop a model Temperature data to Smart Phone using Bluetooth module.<br><br>9. Develop a working model to Switch ON/OFF LED when Commands received as "ON/OFF" String using temperature using Bluetooth.<br><br>10. Develop a working model to store moisture data to any free cloud.<br><br>11. Develop a working model to mark attendance using RFID module and store data in cloud.<br><br>12. Develop a working model to detect gas leakage and send an Sms to a Smart phone(or give a buzzer sound).<br><br>13. Develop a working model to control intensity light using LDR.<br><br>14. Develop a working model to control submersible pump based on Switch Press. | |

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSL71 | CO1 | Understand the concept of Internet of Things |
| | CO2 | Implement interfacing of various sensors with IoT development boards (Arduino and (or) Raspberry Pi) |
| | CO3 | Demonstrate the ability to transmit data wirelessly between different devices |
| | CO4 | Use revelent tools to upload/download sensor data on cloud and server |
| | CO5 | Build IoT devices for day-to-day applications |

| Course Title :**Web Application Security Lab** | | |
|---|---|---|
| Subject Code : **19CSL72** | Credit :**1** | CIE: 50 |
| Number of Practical Hours | 3 | SEE: 50 |
| Total Number of Lecture./Practical Hours | | SEE Hours: 03 |
| Prerequisites: Computer Networks | | |

**Course Objectives:** The objective of the course is to study different tools in security analysis of web applications and perform vulnerability analysis.

**List of Programs**

1. Analyse different encoding (Base64, URL, HTML) and encryption (MD5, SHA1, SHA2 etc) mechanisms used in application.
2. Build a sitemap using the application mentioned for analysis.
3. Experiment to perform web application mirroring using HTTrack.
4. Build a checklist for Authentication and apply on the web application to analyse the outcomes.
5. Build a checklist for Session management and use the same to perform manual checks on the application.
6. Experiment to perform Sessions Hijacking using Web-Goat
7. List Horizontal Controls in the application and bypass the roles based functionalities.
8. Experiment to perform SQL Injection in application using manual and automated tools.
9. Experiment to perform OS Command Injection in application and extend the attack to gain web shell access.
10. Build a checklist for file path traversal attacks to access the server internal files.
11. Experiment to Analyse XML Parsers working in the application using XML External Entities.
12. Program to find Business logic flaws in given application.
13. Program to identify open ports in the IP address.
14. Experiment to bypass file upload validation and gain web shell access to the server.
15. Experiment to perform Local File Inclusion and extend the process for executing remote codes.

Note: For SEE, students will be asked to do similar programs

| **Course outcomes**: On completion of the course, the student will have the ability to: | | |
|---|---|---|
| **Course Code** | **CO #** | **Course Outcome (CO)** |
| **19CSL72** | **CO1** | Analyze web application/protocols from security perspective. |
| | **CO2** | Demonstrate use of tools used in security analysis. |
| | **CO3** | Illustrate flaws in authentication management, session management and vertical access control. |
| | **CO4** | Conduct SQL and OS injection in a ethical manner |
| | **CO5** | Demonstrate file path traversal attack and analyze XML parser. |

| **Course Title : SEMINAR** | | |
|---|---|---|
| Subject Code : **19CSS73** | Credit :2 | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |

**Course Objectives:**

- Identify state of art topic in current trends.
- Perform self-study.
- Comprehend the domain knowledge and organize well documented report and makeoverall presentation.

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CSS73** | **CO1** | Identify current and significant topics focusing to current IT trends |
| | **CO2** | Conduct literature survey to identify ,analyse and discuss selected seminar topics |
| | **CO3** | Present the selected topic with effective communication and presentation skills. |
| | **CO4** | Summarize the work and present future scope |
| | **CO5** | Compile and make technical report. |

| **Course Title : PROJECT PHASE - I** | | |
|---|---|---|
| Subject Code : **19CSP74** | Credit :2 | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |

**Course Objectives:**

- Identify real-world problems by performing the Literature survey
- Awareness of design and proposed methodologies and its analysis
- Design Architectural Models and identity the functional & non functional requirements by all team members
- Prepare quality technical report and present in a well-organized manner

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CSP74 | **CO1** | Apply basic engineering knowledge and identify the problem either individually or as a group |
| | **CO2** | Apply Engineering skills to solve problems of Engineering applications |
| | **CO3** | Evaluate the knowledge of contemporary issues through literature survey and formulate the problems |
| | **CO4** | Design the problem using software methodology. |
| | **CO5** | Prepare a well organized report. |

# CURRICULUM

# FOR B.E. VIII SEMESTER

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING
**(An autonomous college under VTU)**
**KALABURAGI**

**About the institution:** The Hyderabad Karnataka Education (HKE) society founded by Late Shri Mahadevappa Rampure, a great visionary and educationist. The HKE Society runs 46 educational institutions. Poojya DoddappaAppa College of Engineering, Gulbarga is the first institution established by the society in 1958. The college is celebrating its golden jubilee year, setting new standards in the field of education and achieving greater heights. The college was started with 50% central assistance and 50% state assistance, and a desire to impart quality technical education to this part of Karnataka State. The initial intake was 120 with degree offered in three branches of engineering viz, Civil, Mechanical and Electrical Engineering. Now, it houses 11 undergraduate courses, 10 post Graduate courses and 12 Research centers, established in Civil Engg., Electronics & Communication Engg, Industrial & Production Engg, Mechanical Engg, Electrical Engg., Ceramic Cement Tech., Information Science & Engg., Instrumentation Technology, Automobile Engg., Computer Sc. and Engg., Mathematics and Chemistry All the courses are affiliated to Visveswaraya Technological University, Belgaum. At present the total intake at UG level is 980 and PG level 193.

The college receives grant in aid funds from state government. A number of projects have been approved by MHRD /AICTE, Govt. of India for modernization of laboratories. KSCST, Govt. of Karnataka is providing financial assistance regularly for the student's projects.

The National Board of Accreditation, New Delhi, has accredited the College in the year 2005-08 for 09 UG Courses out of which 08 courses are accredited for three years and 01 course is accredited for five years. And second time accredited for Six Course in the year 2009-2012

Our college is one among the 14 colleges selected under TEQIP, sponsored by World Bank. It has received a grant of Rs 10.454 Crores under this scheme for its development. The institution is selected for TEQIP phase II in year 2011 for four years. Institution is receivinga grant of Rs 12.50 Crores under TEQIP Phase -II scheme for its development and selected for TEQIP-III as mentoring Institute for BIET Jhansi(UP).

Recognizing the excellent facilities, faculty, progressive outlook, high academic standards and record performance, the VTU Belgaum reposed abundant confidence in the capabilities of the College and the College was conferred Autonomous Status from the academic year 2007-08, to update its own programme and curriculum, to devise and conduct examinations, and to evaluate student's performance based on a system of continuous assessment. The academic programmers are designed and updated by a Board of Studies at the department level and Academic Council at the college level. These statutory bodies are constituted as per the guidelines of the VTU Belgaum. A separate examination section headed by aController of Examinations conducts the examinations.
At present the college has acquired the Academic autonomous status for both PG and UG courses from the academic year 2007-08 and it is one among the six colleges in the state of Karnataka to have autonomous status for both UG and PG courses.

One of the unique features of our college is, it is the first college in Karnataka State to start the Electronics and Communication Engineering branch way back in the year 1967, to join NIT Surathkal and IISc, Bangalore. Also, it is the only college in the state and one among the three colleges across the country, offering a course in Ceramic and Cement Technology. This is the outcome of understanding by faculty and management about the basic need of this region, keeping in view of the available raw material and existing Cement Industries.

Bharatiya Vidya Bhavan National Award for an Engineering College having Best Overall Performance for the year 2017 by ISTE (Indian Society for Technical Education). In the year 2000, the college was awarded as Best College of the year by KSCST, Bangalore in thestate level students projects exhibition.

The college campus is spread over 71 acres of land on either side of Mumbai-Chennai railway track and has a sprawling complex with gardens and greenery all around.

**About the department:** The Computer Science and Engineering department was started in the year 1984 with an intake of 40 students for UG. The department has seen phenomenal growth and now the department has increased UG intake to 120 students and offering two Post Graduation programmes : PG (Computer Science and Engineering with an intake of 25 students) and PG(Computer Network and Engineering with an intake of 18 students). The department is offering research program under its recognized research center. The department is having state-of-the-art computing facilities with high speed internet facilities and laboratories. The department library provides useful resources like books and journals. The department has well qualified and experienced teaching faculty. The department has been conducting several faculty development programs and student training programs.

## Vision of the Institution

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

## Mission of the Institution

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

## Vision of the Department

- To become a premier department in Computer education, research and to prepare highly competent IT professionals to serve industry and society at local and global levels.

## Mission of the Department

- To impart high quality professional education to become a leader in Computer Science and Engineering.
- To achieve excellence in Research for contributing to the development of the society.
- To inculcate professional and ethical behaviour to serve the industry.

**Program Educational Objectives (PEO)**:

| | |
|---|---|
| **PEO1:** | To prepare graduates with core competencies in mathematical and engineering fundamentals to solve and analyze computer science and engineering problems |
| **PEO2:** | To adapt to evolving technologies and tools for serving the society |
| **PEO3:** | To perform as team leader, effective communicator and socially responsible computer professional in multidisciplinary fields following ethical values |
| **PEO4:** | To encourage students to pursue higher studies, engage in research and to become entrepreneurs |

**Program Outcomes:**

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

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

03. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

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

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

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

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

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

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

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

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

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

**Program Specific Outcomes (PSOs):**

| | |
|---|---|
| **PSO1:** | Acquire competency in hardware and software working principles to analyze and solve computing problems. |
| **PSO2:** | Design quality software to develop scientific and business applications following Software Engineering practices. |
| **PSO3:** | Apply cutting edge technologies using modern tools to find novel solutions ethically to existing problems. |

## SCHEME OF TEACHING FOR VIII SEMESTER - 2022-2023

### B.E. (COMPUTER SCIENCE AND ENGINEERING)

| Code No. | Course | Teaching Hours/Week | | | | Examination | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Lecture | Tutorial | Self Study | Practical | CIE | SEE | Total | Credit |
| 19CS81 | Data Warehousing andMining | 3 | 0 | 1 | 0 | 50 | 50 | 100 | 3 |
| 19CS82X | Professional Elective- 4 | 3 | 0 | 0 | 0 | 50 | 50 | 100 | 3 |
| 19CS8OE | Open Elective- III | 3 | 0 | 0 | 0 | 50 | 50 | 100 | 3 |
| 19CSMC84 | Certification Course (NPTEL/MOOC) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 19CSINT85 | Internship | 4 – 6 Weeks | 0 | 0 | 0 | 50 | 0 | 50 | 2 |
| 19CSP81 | Project Phase - II | 0 | 0 | 0 | 2 | 50 | 50 | 100 | 8 |
| | **Total** | **9** | **0** | 1 | **2** | **250** | **200** | **450** | **20** |

**Professional Elective- 4**

| | |
|---|---|
| 19CS821 | Software Defined Network |
| 19CS822 | Block Chain Technology |
| 19CS823 | Information Storage Management |
| 19CS824 | Big Data Analytics |

**Open Elective- III**

| | |
|---|---|
| 19CS8OE | Fundamentals of Python Programming |

| **Course Title: DATA WAREHOUSING AND MINING** | | | |
|---|---|---|---|
| Subject Code : **19CS81** | Credits: 3 | | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | **Self Study : 1 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | | SEE Hours: 03 |
| **Prerequisite:** Basic Knowledge about Data base, Engineering Mathematics and Statistics**.** | | | |
| **Course objectives:**<br>• Understanding the fundamentals of data mining and useful patterns from random data<br>• Visualizing the information patterns from data collected from various domains<br>• Ability to create predictive models | | | |

| **Modules** | **Teaching Hours** |
|---|---|
| **Module I**<br><br>**Introduction**: Why Data Mining, Kinds of Data Can be Mined, Kinds of Patterns can be Mined, Technologies used for Data Mining, Kinds of Applications Targeted, Major issues in Data Mining. Data Objects and Attribute types, Measuring Data Similarity and Dissimilarity, Data Preprocessing: Data Preprocessing Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization | **09 Hrs** |
| **Module II**<br><br>**Data Warehouse and Online Analytical Processing:** Data Warehouse, Data Warehouse Modeling for Data cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute-Oriented Induction. | **08 Hrs** |
| **Module III**<br><br>**Mining Frequent Patterns, Associations and Correlations:** Basic Concepts, Frequent Itemset Mining Methods, Which Patterns Are Interesting-Pattern Evaluation. Classification Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Bayesian Belief Networks, Lazy Learners. | **09 Hrs** |
| **Module IV**<br><br>**Cluster Analysis: Basic Concepts and Methods** Cluster Analysis, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Evaluation of Clustering, Clustering High-Dimensional Data, Clustering with Constraints, Outliers and Outlier Analysis. | **08 Hrs** |
| **Module V**<br><br>**Data Mining Trends and Research Frontiers:** Mining of Complex Data Types, Methodologies of Data Mining, Data Mining Applications, Data Mining and Society, Data Mining Trends. | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**TEXT BOOKS:**

1. Jiawei Han, Micheline Kamber, Jian Pei "***Data Mining – Concepts and Techniques***" -Morgan Kaufmann Publishers, 3rd Edition, 2012.

**REFERENCES:**

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, "***Introduction to Data Mining***" Pearson education, 2006.
   2003.
2. Arun K Pujari, "***Data Mining Techniques***" –University Press, Private Limited, 2013.
3. C.C. Aggarwal, "***Data Mining"*** Springer International Publishing Switzerland 2016.

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CS81** | **CO1** | Identify the scope and necessity of Data Mining and Warehousing for the Society. |
| | **CO2** | Illustrate the analysis of Data Warehouse and Online Analytical Processing |
| | **CO3** | Design and deploy appropriate classification techniques. |
| | **CO4** | Abilityto develop various algorithms based on Cluster Analysis |
| | **CO5** | Discuss the Data Mining trends and applications. |

| **Course Title: SOFTWARE DEFINED NETWORKS** | | |
|---|---|---|
| Subject Code : 19CS821 | Credits: 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisite:** Computer Networks. | | |
| **Course objectives:**<br>• To understand the separation of Data Plane and Control plane.<br>• To Study SDN applications.<br>• Able to understand Network functions Virtualization and concepts. | | |
| **Modules** | | **Teaching Hours** |
| **Modules I**<br>SDN Background and Motivation : Evolving network requirements-The SDN Approach: Requirements, SDN Architecture, Characteristics of Software-Defined Networking, SDN and NFV-Related Standards: Standards-Developing Organizations, Industry Consortia, Open Development Initiatives | | **09 Hrs** |
| **Module II**<br>SDN Data plane and OpenFlow : SDN data plane: Data plane Functions, Data plane protocols, Openflow logical network Device: Flow table Structure, Flow Table Pipeline, The Use of Multiple Tables, Group Table- OpenFlow Protocol | | **08 Hrs** |
| **Module III**<br>**SDN Application Plane :** SDN Application Plane Architecture: Northbound Interface, Network Applications, User Interface- Network Services Abstraction Layer: Abstractions in SDN, Frenetic- Traffic Engineering Measurement and Monitoring Security- Data Center Networking- Mobility and Wireless. | | **08 Hrs** |
| **Module IV**<br>**Network Functions Virtualization :** Background and Motivation for NFV-Virtual Machines- NFV Concepts: Simple Example of the Use of NFV, NFV Principles, High-Level NFV Framework, NFV Benefits and Requirements- NFV Reference Architecture: NFV Management and Orchestration. | | **08 Hrs** |
| **Module V**<br>Network Functions Virtualization : Background and Motivation for NFV-Virtual Machines- NFV Concepts: Simple Example of the Use of NFV, NFV Principles, High-Level NFV Framework, NFV Benefits and Requirements- NFV Reference Architecture: NFV Management and Orchestration | | **09 Hrs** |
| **Question paper pattern:**<br>The question paper will have ten questions. | | |
| There will be 2 questions from each module, covering all the topics from a module.<br>The students will have to answer 5 full questions, selecting one full question from each module**.** | | |

**TEXT BOOKS:**

1. William Stallings, "Foundations of Modern Networking", Pearson Ltd.,2016.
2. Software Defined Networks: A Comprehensive Approach by Paul Goransson and Chuck Black,Morgan Kaufmann Publications, 2014
3. SDN - Software Defined Networks by Thomas D. Nadeau & Ken Gray, O'Reilly, 2013

**REFERENCES:**

1. Feamster, Nick, Jennifer Rexford, and Ellen Zegura. "The road to SDN: an intellectual history of programmable networks." ACM SIGCOMM Computer Communication Review 44.2 (2014): 87-98.
2. Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76. Online Resources 1. https://www.coursera.org/learn/sdn

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS821 | CO1 | Explain the key benefits of SDN by the separation of data and control planes |
| | CO2 | Interpret the SDN data plane devices and Openflow Protocols |
| | CO3 | Implement the operation of SDN control plane with different controllers |
| | CO4 | Apply techniques that enable applications to control the underlying network using SDN |
| | CO5 | Describe Network Functions Virtualization components and their roles in SDN |

| **Course Title: BLOCKCHAIN TECHNOLOGY** | | |
|---|---|---|
| SubjectCode:**19CS822** | Credits:3 | CIE:50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE:50 |
| Total Number of Lecture Hours | 42 | SEEHours:03 |
| **Pre-Requisite:** Computer Networks, Security Basic Concepts. | | |
| **Course objectives:**<br>● Understand the philosophy of Block chain and the cutting edge technology behind its functions<br>● Illustrate how to setup Ethereum tools<br>● Explain the key vocabulary and concepts used in Block chain for Business. | | |

| **MODULES** | **Teaching Hour** |
|---|---|
| **MODULE-I**<br>**Basics of Block chain**: Introduction, Concept of Block chain, History, Definition of Block chain, Fundamentals of Block chain, Characteristics of Block chain, Consensus in Trust –Building Exercise, Public, Private and Hybrid Block chain, Distributed Ledger Technologies, DLT Decentralized Applications, Architecture of Block chain, Transactions, Chaining Blocks, Value Proposition of Block chain Technology<br><br>**Decentralized System**: Introduction, Distributed Decentralized Databases, Decentralized Enterprise, Decentralization, Disintermediation, Decentralized Enterprise Regulation. | **8 Hrs** |
| **Module– II**<br>**Hash Functions**: Introduction, Hashing, Message Authentication Code, Secure Hash Algorithms (SHA-1), Secure Hash Algorithm Version 3, Distributed Hash Tables, Hashing and Data Structures, Hashing in Blockchain Mining.<br><br>**Consensus:** Introduction, Consensus Approach, Consensus Algorithms, Byzantine Agreement Methods | **8 Hrs** |
| **Module-III**<br><br>**Block chain Components**: Introduction, Ethereum, History, Ethereum Virtual Machine, Working of Ethereum, Ethereum Clients, Cryptography: Introduction, Cryptography and its primitives, Symmetric Cryptography, Asymmetric Cryptography.<br>Smart Contracts: Introduction, Smart Contracts, Absolute and Immutable, Contractual Confidentiality, Law Implementation and Settlement, Characteristics, Internet of Things, Types of Smart Contracts, Types of Oracles. | **8 Hrs** |
| **Module-IV**<br><br>**Consortium Block chain:** Introduction, Key Characteristics of Consortium Block chain, Why we need Consortium Block chain, Hyperledger Platform, Overview of Ripple, Overview of Corda. | **8 Hrs** |

| | |
|---|---|
| **Initial Coin Offering**: Introduction, Block chain Fundraising methods, Launching an ICO, Investing in an ICO, Pros and Cons of Initial Coin Offering, Successful Initial Coin Offerings, Evolution of ICO. | |
| **Module-V**<br>**Security in Block chain:** Introduction, Security Aspects in Bitcoin, Security and Privacy Challenges of Block chain in General, Performance and Scalability, Identity Management and Authentication, Regulatory Compliance and Assurance, Safeguarding Blockchain Smart Contract (DApp), Security Aspects in Hyper ledger Fabric.<br><br>**Applications of Block chain:** Introduction, Block chain in Banking, Block chain in Education, Block chain in Health Care, Block chain in Supply chain, The Block chain and IoT. | **10 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books:**

1. Kumar Saurabh, Ashutosh Saxena, "Blockchain Technology Concepts and Applications", First Edition, Wiley India Pvt, 2020.

**Refer the above mentioned text book for Module I, Module II and Module III.**

2. Chandramouli Subramanian, Asha A George, Abhilash K A and MeenaKarthikeyan, "Block chain Technology", University Press, 2021.

**Refer the above mentioned text book for Module III, Module IV and Module V.**

**Reference Books:**

1. Antonopoulos, Mastering Bitcoin: Unlocking Digital Cryptocurrencies
2. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System
3. DR. Gavin Wood, "ETHEREUM: A Secure Decentralized Transaction Ledger,"Yellow paper.2014.
4. Nicola Atzei, Massimo Bartoletti, and Tiziana Cimoli, A survey of attacks on Ethereum smart contracts

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

| Course Code | CO# | Course Outcome(CO) |
|---|---|---|
| 19CS822 | **CO1** | Understand the concept, fundamentals, Characteristics and definition of Block chain. |
| | **CO2** | Illustrate the use of Hash Functions and Consensus |
| | **CO3** | Experiment with Block chain Components and Smart contracts Examples and Patterns. |
| | **CO4** | Make use of Consortium Block chain and Initial Coin Offering |
| | **CO5** | Develop Security in Block chain and its applications. |

| **Course Title: INFORMATION STORAGE MANAGEMENT** | | |
|---|---|---|
| Subject Code: **19CS823** | **Credits : 3** | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Prerequisites:** DBMS, Computer Networks | | |
| **Course Objectives:**<br>• Storage devices principles including structure, host I/O processing, & corealgorithms<br>• Storage classes (SAN, NAS,CAS), interconnection protocols, and management principles and design principles.<br>• Networked storage capabilities (Snaps, mirroring, virtualization) | | |

| **Modules** | **Teaching Hours** |
|---|---|
| **Module - I**<br>**Storage System:** Introduction to Information Storage: Evolution of Storage Architecture, Data Center Infrastructure, Virtualization and Cloud Computing. Data Center Environment: Application, Host (Compute), Connectivity, Storage. Data Protection: RAID: RAID Implementation Methods, RAID Techniques, RAID Levels, RAID Impact on Disk Performance. Intelligent Storage Systems: Components of Intelligent Storage System, Storage Provisioning. | **08 Hrs** |
| **Module – II**<br>**Storage Networking Technologies:** Fibre Channel Storage Area Networks: Components of FC SAN, FC connectivity, Fibre Channel Architecture, Zoning, FC SAN Topologies, Virtualization in SAN. IP SAN and FCoE: iSCSI, FCIP, FCoE. Network Attached Storage: Components of NAS, NAS I/O Operation, NAS File-Sharing Protocols, File-Level Virtualization, Object-Based Storage and Unified Storage: Object-Based Storage Devices, Content-Addressed Storage, Unified Storage. | **08 Hrs** |
| **Module –III**<br>**Backup, Archive and Replication:** Introduction to Business Continuity: Information Availability, BC Terminology, BC Planning Lifecycle, Failure Analysis, BC Technology Solutions. Backup and Archive: Backup Methods, Backup Topologies, Backup Targets, Data Deduplication for Backup, Backup in Virtualized Environments, Data Archive. Local Replication: Replication Terminology, Uses of Local Replicas, Local Replication Technologies, Local Replication in a Virtualized Environment. Remote Replication: Remote Replication Technologies, Three-Site Replication, Remote Replication and Migration in a Virtualized Environment. | **08 Hrs** |
| **Module -IV**<br>**Cloud Computing and Virtualization:** Cloud Enabling Technologies, Characteristics of Cloud Computing, Benefits of Cloud Computing, Cloud Service Models, Cloud Deployment Models, Cloud Computing Infrastructure, Cloud Challenges and Cloud Adoption Considerations. Virtualization Appliances: Black Box Virtualization, In-Band Virtualization Appliances, Outof-Band Virtualization Appliances, High Availability for Virtualization | **09 Hrs** |

| | |
|---|---|
| Appliances, Appliances for Mass Consumption. Storage Automation and Virtualization: Policy-Based Storage Management, Application-Aware Storage Virtualization, Virtualization-Aware Applications. | |
| **Module - V**<br><br>**Securing and Managing Storage Infrastructure:** Securing and Storage Infrastructure: Information Security Framework, Risk Triad, Storage Security Domains, Security Implementations in Storage Networking, Securing Storage Infrastructure in Virtualized and Cloud Environments. **Managing the Storage Infrastructure:** Monitoring the Storage Infrastructure, Storage Infrastructure Management activities, Storage Infrastructure Management Challenges, Information Lifecycle management, Storage Tiering. | **09 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text Books :**

1. Information Storage and Management, Author :EMC Education Services, Publisher: Wiley ISBN: 9781118094839.

2. Storage Virtualization, Author: Clark Tom, Publisher: Addison Wesley Publishing Company ISBN : 9780321262516

**Reference Books :**

1. Marc Farley Osborne, "Building Storage Networks", Tata McGraw Hill
2. Robert Spalding, "Storage Networks: The Complete Reference", Tata McGraw Hill

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS823 | CO1 | Identify key challenges in managing information. |
| | CO2 | Analyze different storage networking technologies and virtualization |
| | CO3 | Describe important storage technologies features such as availability, replication, scalability and performance |
| | CO4 | Understand the concept of cloud and forms of virtualization. |
| | CO5 | Illustrate the security storage infrastructure and management activities. |

| **Course Title: BIG DATA ANALYTICS** | | |
|---|---|---|
| SubjectCode:**19CS824** | Credits:3 | CIE:50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE:50 |
| Total Number of Lecture Hours | 42 | SEEHours:03 |

**Pre-Requisite:** Data Warehousing, Data mining.

**Course objectives:**

- Understand an overview of Apache Hadoop
- Ability to understand the HDFS Concepts and Interfacing with HDFS
- Understand Map Reduce
- Apply analytics on Structured, Unstructured Data. Exposure to Data Analytics with R.

| **MODULE** | **Teaching Hours** |
|---|---|
| **MODULE-I**<br>**Types of Digital Data** – Classification of Digital Data. Introduction to Big Data – Characteristics of Data, Evolution of Big Data, Definition of Big Data, Challenges with Big Data, What is Big Data? Big Data Analytics – What is Big Data Analytics? Classification of Analytics, Top Challenges Facing Big Data, Data Science, Data Scientist…Your New Best Friend!!!, Terminologies Used in Big Data Environments, Basically Available Soft State Eventual Consistency (BASE). The Big Data Technology Landscape – NoSQL (Not Only SQL), Hadoop. | **09 Hrs** |
| **Module– II**<br>**Introduction to Cassandra:** Apache Cassandra- An Introduction, Features of Cassandra**,** CQL Data Types, CQLSH, Keyspaces, CRUD, Collection, Using a Counter,Time to Live, Alter Commands, Import and Export, Querying System Tables. | **09 Hrs** |
| **Module-III**<br>**HDFS:** Goals and Motivation, Design, Daemons, Reading and Writing Data, Managing Filesystem Metadata, Namenode High Availability, Namenode Federation, Access and Integration, Command-Line Tools.<br>MapReduce: The Stages of MapReduce, Introducing Hadoop Mapreduce, Daemons, When it All Goes Wrong, YARN. | **08 Hrs** |
| **Module-IV**<br>**Identity, Authentication, and Authorization:** Identity, Kerberos and Hadoop, Kerberos and Hadoop, Kerberos: A Refresher, Kerbros Support in HAdoop, Authorization, HDFS, Mapreduce, Other Tools and Systems, Tying It Together.Resource Management: What is Resource Management, HDFS Quotas, Mapreduce Schedulers, The FIFO Schedulers, The Fair Scheduler, The Capasity Scheduler, The Future. | **08 Hrs** |

| **Module-V** | |
|---|---|
| **Cluster Maintenance:** Managing Hadoop Processes, HDFS Maintenance Tasks, MapReduce Maintenance Tasks. Troubleshooting: Differential Diagnosis Applied to Systems, Common Failures and Problems, "Is the Computer Plugged In?", Treatment and Care. | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
Therewillbe2questions from each module, covering all the topics from a module.
The students will have to answer 5fullquestions, selecting one full question from each module**.**

**Text Books:**
1. Seema Acharya, Subhashini Chellappan, Big data and Analytics, Wiley publications, 2014. 2. Tom White, Hadoop: The Definitive Guide, Fourth Edition, O'Reilley, 2015.
**Refer the above mentioned text book for Module 1 & Module 2.**
2. 1. Eric Sammer, Hadoop Operations, O'Reilley, 2012. 2. Vignesh Prajapati, Big data analytics with R and Hadoop, SPD 2013.
**Refer the above mentioned text book for Module 3, 4 & 5**

**Reference Books:**
1. Tom White, "Hadoop: The Definitive Guide", 4th Edition, O"Reilly Media, 2016. 2. Boris Lublinsky, Kevin T.Smith, Alexey Yakubovich, "Professional Hadoop Solutions", 1st Edition, Wrox Press, 2014I. 3. Eric Sammer, "Hadoop Operations: A Guide for Developers and Administrators",1st Edition, O'Reilly Media, 2012.

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

| Course Code | CO# | Course Outcome(CO) |
|---|---|---|
| **19CS824** | **CO1** | Identify and understand various types of Digital data and necessity of Big Data Analytics for society |
| | **CO2** | Create and implement Query using Cassandra |
| | **CO3** | Infer tools for the HDFS and Understand Mapreduce concept |
| | **CO4** | Ability to develop various algorithms for Identifying, Authentication, Authorization and Resource Management |
| | **CO5** | Illustrate how to Maintain Cluster and Trouble Shooting in Big Data Analytics |

| Course Title: **FUNDAMENTALS OF PYTHON PROGRAMMING** | | |
|---|---|---|
| Subject Code : **19CS8OE** | Credit : 3 | CIE: 50 |
| Number of Lecture Hours/Week | **3 Hrs** | SEE: 50 |
| Total Number of Lecture Hours | 42 | SEE Hours: 03 |
| **Pre-requisites:** Knowledge about programming languages like C or C++ and Oops concepts**.** | | |
| **Course objectives:**<br>Learn python program with perspective of handling large amount of data with python data structures much better than C language.<br>• Understand the applications of python language in various domains of applications.<br>• Develop database applications in python programming language. | | |
| **MODULES** | | **Teaching Hours** |
| **Module –I**<br><br>**Py Ingredients:** Introduction-Python in the Real World, Python versus Language X. Python2, Python3, installation of python.<br>Numbers, Strings, constants and variables: Variables, Names and Objects, Numbers, Strings. | | **08 Hrs** |
| **Module-II**<br><br>**Py Filling: Lists, Tuples, Dictionaries, and Sets:** Lists, Tuples, Dictionaries, Sets.<br>Python Code Structures: Comment with #, Continue Lines with \, Compare with if, elif, and else, Repeat with while, Iterate with while, Iterate with for, Comprehensions, Functions, Generators, Decorators, Namespaces and scope, Handle errors with try and except, Make own exceptions. | | **09 Hrs** |
| **Module-III**<br>**Py Boxes: Modules, Packages, and Programs: Stand** alone programs, command-line arguments, Modules and import statement, Packages, Python standard library.<br>More batteries: Get other python code.<br>**Objects and Classes:** Define class with class, inheritance, Override a method, Add a method, Get help with your parent with super, in self Defense , Get and Set Attributes, Values with properties, Name Managing for privacy, Method Type Duck Typing, Special Methods | | **09 Hrs** |
| **Module- IV**<br>**Mangle Data like a Pro:** Text Strings, Binary Data. File Input/output, Structured Text files, Structured Binary Files, Relational Databases, NoSQL Data Stores, Full-Text Databases. | | **08 hrs** |
| **Module – V**<br><br>**The Web, Untangled, Concurrency and Networks:** Web Clients, Web servers, Web services and Automation. **Systems:** Files, Directories, Programs and Processes, Calendars and Clocks. | | **08 Hrs** |

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module, covering all the topics from a module.
The students will have to answer 5 full questions, selecting one full question from each module**.**

**Text books:**

1. Introduction to Python – Modern computing in simple packages- Bill Lubancovic O‟Reilly publication , Jan 2018.
   www.python.org , www.w3resource.com , www.tutorialpoint.com .

**Reference Books:**

1. Allen Downey , Jeffrey Elkner,Chris Meyrs "How to Think like a Scientist- Learning with Python" -,Ggreen tea press 2002 First Edition.
2. "Introduction to Computer Science using Python" – Charles Dierbach, Wiley publication, Dec 25th 2012
3. Beginning Python – from Novice to Professional – Magnus Lie Hetland, 2nd Edition. Apress publication, 3rd Nov 2008

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| 19CS8OE | **CO1** | Demonstrate the understanding and usage of python scripting elements like constructs, data structures, functions, modules, packages and regular expressions. |
| | **CO2** | Distinguish between the data structure of python and other languages. |
| | **CO3** | Translate the given algorithm into a python program. |
| | **CO4** | Demonstrate handling large volume of data using built in library functions of python. |
| | **CO5** | Develop real world applications using Files, Directories, Programs and Processes, Calendars and Clocks. |

| **Course Title : CERTIFICATION COURSE(NPTEL/MOOCS)** | | |
|---|---|---|
| Subject Code : **19CSMC84** | Credit :1 | CIE: 50 |
| | | SEE: 50 |
| | | SEE Hours: 03 |

**Course Objectives:**

To enable the students to get exposure to Recent trends in the field related to Computer Science & Engineering.

Every student should undergo National Programme on Technology Enhanced Learning (NPTEL) Online certification course for the duration of 4 Weeks to 8 Weeks. For more details on these online courses under NPTEL you may visit the link http://onlinecourses.nptel.ac.in which are similar to the MOOCS offered in platforms like edX, Coursera etc can be offered under this initiative.

| **Course Title : INTERNSHIP** | | |
|---|---|---|
| Subject Code : **19CSINT85** | Credit :1 | |
| CIE: 50 | SEE: 50 | |
| | | |

**Course Objectives:**

This course will enable students to get the field exposure and experience

Internship shall be taken up on a topic relevant to computer science and engineering and use the skill set developed through second and third year degree courses.

Student shall finalize the topic in consultation with Internship coordinator and get the topic approved by internship coordinator/internship committee/HOD.

The industry/organization should issue certificates of internship offer and its completion. The offer letter should clearly have the nature of work to be done by the student and the supervisor‟s name and duration of internship.

The internship should be completed during vacation after VII and VIII Semester.

Duration of the Internship shall be 4 - 6 Weeks. Student shall make a midterm and final presentation of the activities undertaken during the first 3 - 4 weeks (In last week of VII sem or at the beginning of VIII sem) and at the end of 6th week of internship (preferably latest by last week of VIII sem) respectively, to a panel comprising internship co-ordinator, internship guide and head of the department. Each student should submit the internship report at the end of semester with internship certificate. Viva-voce examination shall be conducted by a panel of examiners.

| **Course Title : PROJECT PHASE - II** | | |
|---|---|---|
| Subject Code : **19CSP81** | Credit :10 | CIE: 50 |
| Number of Practical Hours/Week | **2 Hrs** | SEE: 50 |
| | | SEE Hours: 03 |

**Course Objectives:**

Enable students to

- Design functional modules and develop the module using state of the technologies.
- Implement functional modules using advance tools and techniques
- Work effectively as a team
  Prepare well documented report

**Course outcomes:**

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

| Course Code | CO # | Course Outcome (CO) |
|---|---|---|
| **19CSP81** | **CO1** | Perform self study and exhibit the skills of self learning by demonstrating sound technical knowledge on the topic selected for project work |
| | **CO2** | Execute the selected task with team work as per the plan and schedule demonstrating ethics and professional responsibility |
| | **CO3** | Design solution to selected complex engineering problem using modern tools and provide reasonably acceptable solution to satisfy desired goals, and provide reasonably acceptable solution to satisfy desired goals, and environmental sustainability |
| | **CO4** | Prepare a well organized and compiled thesis |
| | **CO5** | Communicate technical results, information and conclusions to others by means of formal presentations |