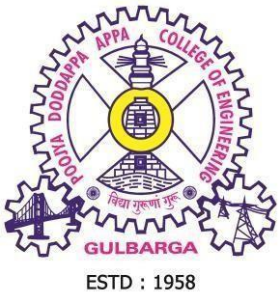
CURRICULUM

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(B.E in COMPUTER SCIENCE & ENGINEERING)

21 Series



POOJYA DODDAPPA APPA COLLEGE OF ENGINEERING

(An autonomous college under VTU)

KALABURAGI

About the institution: The Hyderabad Karnataka Education (HKE) society founded byLate 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 intakewas 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.

\

Hyderabad Karnataka EducationSociety's

PoojyaDoddappaAppa Engineering College,Kalaburagi (An Autonomous Institution)

Aiwan-E-Shahi Area, Kalaburagi, Karnataka 585102

Department of Computer Science & Engineering

SCHEMEOFTEACHINGFORIII SEMESTER–21SERIES

Sl. No	CourseCode	CourseTitle	TeachingHours/Week				Examination				Credits
			TheoryLecture(L)	Tutorial(T)	Practical	Self Study(S)	Duration inhours	CIEMarks	SEEMarks	TotalMarks	
1	21MA31D	Computational Methods for Computer Science	3	0	0	0	3	50	50	100	3
2	21CS32	Mathematical Foundations of Computer Science	3	0	0	0	3	50	50	100	3
3	21CS33	Data Structures	3	0	0	0	3	50	50	100	3
4	21CS34	Microprocessors and Microcontrollers	3	0	0	0	3	50	50	100	3
5	21HU35	Constitution of India & Professional Ethics	2	0	0	0	3	50	50	100	1
6	21INT36	Summer Internship-I	0	0	0	0	0	50	0	50	2
7	21CSAE36A	HTML and CSS	0	0	2	0	3	50	50	100	1
8	21UHV36B	Universal Human Values –I	0	2	0	0	2	50	50	100	1
9	21CSL31	Logic Design Lab	0	0	2	0	3	50	50	100	1
10	21CSL32	Data Structures Lab	0	0	2	0	3	50	50	100	1
11	21CSL33	Microprocessors and Microcontrollers Lab	0	0	2	0	3	50	50	100	1
		Total	**14**	**2**	**8**	**0**	**29**	**550**	**500**	**1050**	**20**

SCHEMEOFTEACHINGFORIVSEMESTER–21 SERIES

Sl.No.	Course Code	CourseTitle	TeachingHours/Week				Examination				Credits
			Theory Lecture(L)	Tutorial(T)	Practical	Self Study(S)	Duration inhours	CIE Marks	SEEMarks	TotalMarks	
1	21MA41D	Applied Statistics	3	0	0	0	3	50	50	100	3
2	21CS42	Finite Automata And Formal Language	3	0	0	0	3	50	50	100	3
3	21CS43	Analysis and Design of Algorithms	3	0	0	0	3	50	50	100	3
4	21CS44	Object Oriented Programming with JAVA	3	0	0	0	3	50	50	100	3
5	21KAK45	Kannada (Samskrutika)	2	0	0	0	1.5	50	50	100	1
6	21KAN45	Kannada(Balake Kannada)									
7	21CSAE46A	Organic Farming: Horticulture	2	0	0	0	2	50	50	100	2
8	21CSAE46B	MS Office Tools	0	0	2	0	3	50	50	100	1
9	21UHV46C	Universal Human Values-II	0	2	0	0	3	50	50	100	1
10	21CSL41	Analysis and Design of Algorithms Lab	0	0	2	0	3	50	50	100	1
11	21CSL42	Object Oriented Programming with JAVA Lab	0	0	2	0	3	50	50	100	1
12	21CSL43	Web Application Development Lab	0	0	2	0	3	50	50	100	1
		Total	**18**	**0**	**08**	**0**	**28.5**	**550**	**550**	**1100**	**20**

SCHEMEOFTEACHINGFORVSEMESTER–21 SERIES

Sl. No	CourseCode	CourseTitle	TeachingHours/Week				Examination				Credits
			TheoryLecture(L)	Tutorial(T)	Practical	Self Study(S)	Duration inhours	CIEMarks	SEEMarks	TotalMarks	
1	21CS51	Software Engineering and Tools	3	0	0	0	3	50	50	100	3
2	21CS52	Computer Networks	3	0	2	0	3	50	50	100	4
3	21CS53	Operating System	3	0	0	0	3	50	50	100	3
4	21CS54	Database Management System	3	0	0	0	3	50	50	100	3
5	21CSL55	Database Management System Lab	0	0	2	0	3	50	50	100	1
6	21RMI56	Research Methodology & Intellectual PropertyRights	2	0	0	0	3	50	50	100	2
7	21CIV57	Environmental Studies	0	2	0	0	2	50	50	100	1
8	21CSAE581	Python Programming	0	0	2	0	3	50	50	100	1
		Total	**14**	**2**	**6**	**0**	**23**	**400**	**400**	**800**	**18**

S.No.	Course Code	Course Title
1	21CSAE581	Python Programming(Ability Enhancement Course)

SCHEMEOFTEACHINGFORVISEMESTER–21 SERIES

Sl. No	CourseCode	CourseTitle	TeachingHours/Week				Examination				Credits
			TheoryLecture(L)	Tutorial(T)	Practical	Self Study(S)	Duration inhours	CIEMarks	SEEMarks	TotalMarks	
1	21HU61	Entrepreneurship, Management and Finance	3	0	0	0	3	50	50	100	3
2	21CS62	Computer Graphics and Fundamentals of Image Processing	3	0	2	0	3	50	50	100	4
3	21CS63	Artificial Intelligence and Machine Learning	3	0	0	0	3	50	50	100	3
4	21CS64x	Professional Elective -I	3	0	0	0	3	50	50	100	3
5	21CS65OEx	Open Elective-I	3	0	0	0	3	50	50	100	3
6	21CSL66	Artificial Intelligence and Machine Learning lab	0	0	2	0	3	50	50	100	1
7	21CSMP67	Mini Project	0	0	2	0	0	50	0	50	2
8	21INT68	Innovation/Entrepreneurship/Societal Internship (To be carried during intervening period of IV and V semester)	0	0	0	0	0	50		50	3
		Total	**15**	**0**	**6**	**0**	**18**	**400**	**300**	**700**	**22**

Professional Elective-I	
21CS641	System Software and Compiler Design
21CS642	Design of IoT System
21CS643	Cryptography and Information security

Open Elective-I	
21CS65OE1	Introduction to Artificial Intelligence

SCHEMEOFTEACHINGFORVIISEMESTER–21 SERIES (TENTATIVE)

Sl. No	CourseCode	Course Title	TeachingHours/Week				Examination				Credits
			TheoryLecture(L)	Tutorial(T)	Practical	Self Study(S)	Durationinhours	CIEMarks	SEEMarks	TotalMarks	
1	21CS71	Cloud Computing	3	0	0	0	3	50	50	100	3
2	21CS72x	Professional Elective –II	3	0	0	0	3	50	50	100	3
3	21CS73x	Professional Elective –III	3	0	0	0	3	50	50	100	3
4	21CS74OEX	Open Elective –II	3	0	0	0	3	50	50	100	3
5	21CSP75	Project Work	0	0	2	0	3	50	50	100	10
6	21CS76	Ability Enhancement Course (Online- 8 weeks NPTEL)	0	0	0	0	3	50	50	100	2
		Total	**12**	**0**	**2**	**0**	**18**	**300**	**300**	**600**	**24**

ProfessionalElective–II	
21CS721	Web Application Security
21CS722	WirelessNetworks&Mobile Computing
21CS723	Data Mining and Warehousing

OpenElective Course -II	
21CS74OE1	WebTechnologies

Professional Elective–III	
21CS731	BlockchainTechnology
21CS732	BigDataAnalytics
21CS733	Parallel Computing

SCHEMEOFTEACHINGFORVIIISEMESTER–21 SERIES (TENTATIVE)

Sl. No	CourseCode	CourseTitle	TeachingHours/Week				Examination				
			TheoryLecture(L)	Tutorial(T)	Practical	Self Study(S)	Duration inhours	CIEMarks	SEEMarks	TotalMarks	Credits
1	21CS81	Technical Seminar	0	0	0	0	0	100	0	100	1
2	21INT82	Research/ Industry Internship	0	0	0	0	3	100	100	200	15
3	21NS39	NSS (National Service Scheme)	Completed during the intervening period of III semester to VIII semester				0	50	50	100	0
4	21PE39	PE (Physical Education)									
5	21YO39	Yoga									
		Total					**3**	**250**	**150**	**400**	**16**

AUTONOMOUS SYLLABUS FOR B.E III SEMESTER

Course Title: COMPUTATIONAL METHODS FOR COMPUTER SCIENCE		
Subject Code : **21CS31**	Credit : 03	CIE: 50
Number of Lecture Hours/Week	**3 (L)**	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 engineeringfields • Methods of least squares to fit straight line and second degree parabola • Solve the problems using probability theory		
MODULES		**Teaching Hours**
Module I **Finite differences: (**Forward and Backward differences), Interpolation, Newton‟s Forward and Backward formulae. Lagrange‟s interpolation and inverse interpolation formulae. **Numerical differentiation:** Numerical differentiation using Newton‟s forward and backward interpolation formulae and problems. **Numerical integration**: Trapezoidal rule, Simpsons 1/3rd and 3/8th rule, Weddle‟s rule (all formulae and rules without proof)		**6 hours**
Module II **Difference equations and Z-Transforms :** 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 **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 **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 **Time Series and Forecasting:** Moving averages, smoothening of curves,		

forecasting models and methods, Statistical Quality Controls methods.	**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
2. Engineering Mathematics by N. P. Bali and Manish Goyal. Laxmi publications, latest edition
3. Integral Transforms in Science and Engineering- by Kurt Bernado Wolf-springer Publications.

REFERENCES BOOKS:

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

Course outcomes:

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

Course Code	CO #	Course Outcome (CO)
21CS31	**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 : 21CS32	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: To develop mathematical thinking and problem-solving skills associated with writing proofs. To expose students to a wide variety of mathematical concepts those are used in the Computer Science discipline.		
MODULES		**Teaching Hours**
Module I **Mathematical logic**: Basic Connectives and truth tables, Logic Equivalence-The Laws of logic, Logical Implications-Rules of Inference. **Counting:** Permutations, combination, Pigeonhole, Principles. **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 **Function:** Function, Function for Computer Science, Growth of functions, Permutation Functions **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 **Introduction to Graph Theory-I:** Definition & Examples, Sub-graph, complements and graph Isomorphism, Vertex degree, Euler trails and circuits. **Graph Theory-II:** Planar graphs, Hamilton paths and cycles, Graph coloring, chromatic polynomials, Transport networks. (Problem solving using C)		**08 hrs**
Module IV **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. **Languages and finite state machines:** Languages, representations of special grammars and languages, finite state machine, semi groups machines and Languages		**08 hrs**
Module V **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		**09 hrs**

generator matrices, **Groups coding:** coding with coset headers and hamming matrices. Decoding in cosets: the cycle index, polys method of enumeration.	

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 and Ross, "Discrete Mathematical Structures for Computer Science", 5th edition , PHI, 2000 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 Hill2007.

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

Course Code	CO #	Course Outcome (CO)
21CS32	**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
	CO4	Illustrate problems on trees and understand its properties and Design grammars, finite state machines.
	CO5	Demonstrate knowledge of algebraic structures and their applications in coding theory and group coding

Course Title: DATA STRUCTURES		
Subject Code : 21CS33	Credits :3	CIE: 50
Number of Lecture Hours/Week	3 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, hashtables, 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 **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. **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. **Dynamic memory allocation:** Meaning of dynamic memory allocation, MALLOC, CALLOC, Free and REALLOC functions, Pointer revisited. **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	**08 Hrs**
Module - II **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. **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.	**08 Hrs**
Module – III **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. .	**08 Hrs**

Module - IV **Other list structures:** Circular lists, Stack as circular list, Queues as a circular list, Primitive operations on circular list, doubly linked list. **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. **Trees and their applications:** C representation of trees, Tree traversals, General expression as trees, Evaluating an expression tree, Constructing a tree.	**09 Hrs**
Module - V **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. **Hashing:** Resolving hash clashed by open addressing, Choosing a hash function.	**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. 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 APseudocode Approachwith C", Cengage Learning, 2005.
3. Robert Kruse & Bruce Leung, "Data Structures & ProgramDesign 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)
21CS33	**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 ofapplications.
	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: **MICROPROCESSOR AND MICROCONTROLLER**		
SubjectCode:**21CS34**	Credit:3	CIE:50
Number of Lecture Hours/Week	**03 Hrs**	SEE:50
Total Number of Lecture Hours	42	SEEHours:03

Pre-requisites: Logic Design, Basic Electronics

Course Objectives:

- Explore the microprocessor architecture and its instruction set.
- Develop skills for programming in Assembly language.
- Interface Peripheral devices with 8086 Microprocessor and ARM Processor

Modules	Teaching Hours
Module-I **The 8086/8088 Processors :** Architecture of 8086 microprocessor, Signal Descriptions of 8086, Physical Memory Organization, Minimum and Maximum Mode 8086 System and Timings, The Processor 8088. **8086/8088 Instruction Set Assembler Directives :** Machine Language Instruction Formats, Addressing Modes of 8086, Instruction Set of 8086/8088, Machine language Conversion, Assembler Directives and Operation.	**08 Hrs**
Module-II **Assembly Language Programming with 8086/8088:** A Few Machine Level Programs, Machine Coding The Programs, Programming with an Assembler, Assembly Language Example Programs. **Special Architectural Features and Related Programming:** Introduction to stack, stack structure of 8086/88, interrupts and interrupt service routines, Interrupt cycle of 8086/88, Non maskable interrupt, Maskable interrupt, Interrupt programming, passing parameter to procedures, MACROs, Timings and Delays.	**08 Hrs**
Module-III **Special Architectural Features and Related Programming Cont..:** passing parameter to procedures, MACROs, Timings and Delays. **Basic Peripherals and their Interfacing with 8086/88:** Semiconductor Memory interfacing, Dynamic RAM interfacing, Interfacing I/O ports, P/O 8255, Modes of operations of 8255. Interfacing Analog to digital Converter, Interfacing Digital to Analog Converter, Stepper Motor interfacing	**09 Hrs**
Module-IV **Microcontrollers** - Types of Microcontrollers - Criteria for selecting a microcontroller - Example Applications. Characteristics and Resources of a microcontroller. Organization and design of these resources in a typical microcontroller - 8051. 8051 Architecture, Register Organization, Memory and I/O addressing, Interrupts and Stack.	**08 hrs**

Module-V	09 hrs
8051 Addressing Modes, Different types of instructions and Instruction Set, Simple programs. Peripheral Chips for timing control - 8254/8253.**ARMProcessorFundamentals**:Registers , Current Program Status Register , Pipeline, Exceptions, Interrupts, and the Vector Table ,Core Extensions, **Introduction to the ARM Instruction Set :** Data Processing Instructions ,Branch Instructions, Software Interrupt Instructions, Program Status RegisterInstructions,CoprocessorInstructions,LoadingConstants,Simpleprogramming exercises.	

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.

Textbooks:

1. Bhurchandi and Ray, Advanced Microprocessors and Peripherals, Third Edition McGraw Hill, 2012
2. Raj Kamal, Microcontrollers: Architecture, Programming, Interfacing and System Design, Pearson Education, 2011.

ReferenceBooks:

1. Barry B. Brey, The Intel Microprocessors – Architecture, Programming and Interfacing, Eigth Edition, Pearson Education, 2015
2. A. NagoorKani, Microprocessors and Microcontrollers, Second Edition, Tata McGraw Hill, 2012.

Course outcomes:

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

Course Outcome	CO#	Course Outcome(CO)
21CS34	**CO1**	Analyze the 8086 processor Structure, Assembly Langua ge 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 x86family
	CO4	Compare Microprocessor and Microcontroller, Explain interfacing through ARM processor, interrupt routines
	CO5	Demonstrate Instruction set and develop programs using ARM processor

Course Title:	**DATA STRUCTURES LAB**	
Subject Code : 21CSL35	Credits : 1	CIE: 50
Number of Practical Hours/Week	**2 Hrs**	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.

1. Design, Develop and Implement a menu driven Program in C for the following 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 the following operations on Strings
   a. Read a Main String (STR), a Pattern String (PAT) and a Replace 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"t use 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 above operations.

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 following Stack Applications
   a. Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^
   b. Solving Tower of Hanoi problem with ndisks

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 above operations.

7. Design, Develop and Implement a menu driven Program in C for the following operations on Singly Linked List (SLL) of integervalues
   a. Create a SLL of N integers byusing 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 Singly Linked List (SSL) of a given integers.

9. Design, Develop and Implement a menu driven Program in C forthe following 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 Priorityqueue.

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
d. 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 maintained in 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 that uses 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:
On completion of the course, the student will have the ability to:

Course Code	CO #	Course Outcome (CO)
21CSL35	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: MICROPROCESSORS AND MICROCONTROLLERS LAB		
Subject Code: 21CSL36	Credits:1	CIE:50
Number of Practical Hours/Week	**2 Hrs**	SEE:50
		SEEHours:03
Prerequisite: C Programming		
CourseObjectives: Explorethemicroprocessorarchitectureandinstructionset		

List of Programs

SOFTWAREPROGRAMS:PARTA

1. Design an ALP to separate even and odd numbers from an array.
2. Design an ALP to find Factorial of a given 8-bit number.
3. Design an ALP to convert 8 bit binary number to its BCD equivalent
4. Design an ALP to generate first 'n' Fibonacci series.
5. Design an ALP to count the number of 0's and 1's in a given number.
6. Design an ALP to create a file and delete an existing file.
7. Design an ALP to display the list of alphabets on the screen.
8. 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.
9. Design and develop an assembly program to sort a given set of "n" 16-bitnumbersinascendingorder.AdoptBubbleSortalgorithmtosortgiveneleme nts.
10. Develop an assembly language program to reverse a given string and verify whether it is a palindrome or not. Display the appropriate message.
11. Develop an assembly language program to compute nCr using recursive procedure. Assume that "n" and "r" are non-negative integers.
12. 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.

HARDWAREPROGRAMS:PARTB

13. Design anddevelopanassemblyprogramtointerface4*4matrixkeyboard. Using ARM TTDMI/LPC2148.
14. Design and develop an assembly program to implement the buzzer using ARMTTDMI/LPC2148

15. 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.

16. Design and develop an assembly language program to
    a. GeneratetheSineWaveusingDACinterface(TheoutputoftheDA Cistobedisplayed on the CRO).
    b. Generate a Half Rectified Sine wave form using the DAC interface.) using ARMTTDMI/LPC2148.

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

Study Experiments:

1. InterfacingoftemperaturesensorwithARMfreedomboard(oranyotherARM microprocessor board)and display temperature on LCD
2. To design ARM cortex based automatic number plate recognition system
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, KEILIDE and Proteus for simulation

Course outcomes:

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

Course Code	CO#	Course Outcome(CO)
21CSL36	CO1	Develop ALP for searching and sorting using 8086 microprocessor.
	CO2	Design and develop assembly programs using 8086 DOS functions, subroutines and micros in assembly language
	CO3	Design and interface of different peripherals with ARM microcontroller.
	CO4	Develop ARM interfacing software for motor and LCD display
	CO5	Construct different wave forms using interfacing 08086 microprocessor

Course Title: LOGIC DESIGN LAB		
Subject Code : 21CSL37	Credits : 1	CIE: 50
Number of Practical Hours/Week	**2 Hrs**	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 ofsensitive 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 UTP and 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 Astable multivibrator circuit using 555 timerfor a given frequency and duty cycle.
4. Design and implement Half adder, Full Adder, Half Subtractor, Full Subtractor using basic gates.
5. Given a 4-variable logic expression, simplify it using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC.
6. Design and implement code converter I)Binary to Gray (II) Gray to Binary Code using basic gates.
7. Design and verify the Truth Table of 3-bit Parity Generator and4-bit Parity Checker using basic Logic Gates with an even parity bit.
8. Realize a D,T,JK Flip-Flop using NAND gates and verify its truthtable.
9. Design and implement a mod-n (n<8) synchronous up counter using JK Flip Flop ICs and Demonstrate its working
10. Design and implement an Asynchronous counter using decade counter IC 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 to DAC 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)
21CSL37	**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: HTML and CSS (Ability Enhancement Course)		
Subject Code :**21CSL39A**	Credit :1	CIE: 50
Number of Practical Hours/Week	**2Hrs**	SEE: 50
		SEE Hours: 03

1. HTML-Introduction, Coding syntax, Document structure, basic HTML code. Create a basic HTML document, inserting a sentence / multiple sentences (For example: Welcome to my website) in between the tags.

2. HTML – Basic Formatting Tags: Paragraphs, Line Breaks, Headings, Horizontal Rules.
Create a HTML document with paragraph, Make different Levels Headings, use line breaks. Insert a horizontal rule between the heading and the paragraphs.

3. Simple Text Effects and Lists: font family, font face, font size, font color, Underline, Struck through, Bold, Italic, Ordered Lists, Unordered Lists, Definition Lists.
Create a HTML document with three different lists with text effects. There should be a minimum of 3 entries in each one.

4. Image Effects: Image formats, Inserting Images, Alt Tag, width and height tags, Alignment, Borders and Spacing
Create a html document and insert image within it using "relative" linking, add alt text to the image, also insert some text after image and top align the text. Put a border around the image (size of your choice) and also spacing around the image (size of your choice).

5. File Management: Linking Text, Linking Images, Embedding Other Media.
Create a HTML document to create a Home page having three links: About College, Departments and Subjects. Create separate web pages for the three links.

6.T ables: Basics, Table border, Table header, Cellspacing, Cellpadding, Cell spanning, Create a HTML document to display your education details in a tabular format.

7. HTML forms : <form>, <input>, <textarea>, <select>, <option>, <fieldset>
Create a HTML document which shows a feedback and newsletter sign-up form.

8. HTML frames : <frame>, <frameset>, <iframe>
Create a HTML document to create a frameset having header, navigation and content sections.

9. Introduction to CSS: Introduction, applying CSS, selectors, coding syntax, text and background properties.
Create a HTML document which creates a simple web page that is styled using CSS.

10. Website designing (Assignment / OEE).

Course Title: APPLIED STATISTICS		
Subject Code : **21MA41D**	Credit : 03	CIE: 50
Number of Lecture Hours/Week	**3Hrs (L)**	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 1. Probability distribution of discrete and continuous random variables 2. Joint probability distributions and discrete and continuous random variables and Morkov‟s chains 3. Analyse the sample data using Large sample test, t-distribution and chi- distribution		
MODULES		**Teaching Hours**
Module I **Probability distributions:** Random variable (Discrete and continuous) p.d.f., c.d.f., Binomial distribution,Poisson distributions, Normal distribution and problems		**6 hours**
Module II **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 **Markov chains: Introduction probability vectors** stochastic matrices, higher transition probability. Stationary distribution of regular Markov chains and absorbing states		**5 hours**
Module IV **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 attributes. And problems**.**		**6 hours**
Module V **Distances in Classification:** Introduction, Euclidean Distance, Manhattan Distance, Euclidean vs Manhattan Distance, Chebyshev Distance, Hamming Distance, Distance calculation 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)
21MA41D	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: **FINITE AUTOMATA AND FORMAL LANGUAGE**		
Subject Code :**21CS42**	Credit : 3	CIE: 50
Number of Lecture Hours/Week	**03 Hrs (L)**	SEE: 50
Total Number of Lecture Hours	42	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 **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.	**09 Hrs**
Module-II **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.	**08 Hrs**
Module-III **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.	**08 Hrs**
Module-IV **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.	**09 hrs**
Module-V **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. **Undecideability: A** Language that is not recursively enumerable; An Undecidable problem that is RE; Post's Correspondence problem; Other undecidable problems.	**08Hrs**

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)
21CS42	CO1	Design Deterministic and non Deterministic finite automata for a givenlanguage 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: ANALYSIS AND DESIGN OF ALGORITHM		
Subject Code : **21CS43**	Credit : 3	CIE: 50
Number of Lecture Hours/Week	**03 Hrs (L)**	SEE: 50
Total Number of Lecture Hours	**42**	SEE Hours: 03
Pre-requisites: Data structures using C.		
Course objectives: • Analyze the asymptotic performance of the algorithms in time and space domain. • Introduce various algorithm design techniques.		

Modules	**Teaching Hours**
Module-I 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.	**08 Hrs**
Module-II **Brute Force:** Introduction, Selection sort, Bubble Sort, Sequential search Brute-Force String Matching Exhaustive Search **Divide & Conquer :** Introduction, Merge Sort, Quick Sort, Binary Search, Binary tree traversals & related properties, Multiplication of large integers & Stressen's Matrix Multiplication Insertion Sort.	**09 Hrs**
Module-III **Decrease & Conquer** : Introduction, Depth First search, Breadth First Search, Topological Sorting, Algorithms for Generating Combinatorial objects. **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.	**09 Hrs**
Module-IV **Dynamic Programming:** Introduction, Computing a Binomial Coefficient, Warshall's Algorithm, Floyd's Algorithm, The Knapsack Problem and Memory Functions. **Greedy Techniques**: Introduction, Minimum Spanning Tree, Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffmancodes.	**08 Hrs**
Module-V **Limitations of Algorithms Power:** Introduction, Lower- Bound Arguments, Decision Trees, P, NP, and NP – Complete Problems. **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.	**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. 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)
21CS43	CO1	Study the 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: JAVA PROGRAMMING		
Subject Code : **21CS44**	**Credit : 03**	CIE: 50
Number of Lecture Hours/Week	**04 Hrs**	SEE: 50
Total Number of Lecture Hours	42	SEE Hours: 03
Prerequisites: Concepts of Object oriented programming		
Course Objectives: Learn the Java Programming to develop applications, create and import packages, creating GUI with applets, and establish JDBC-ODBC connectivity.		
MODULES		**Teaching Hours**
Module I **The Java Language** -The History and Evolution of Java , An Overview of Java , Data Types, Variables, and Arrays ,Operators , Control Statements **,** A first simple program **Stringhandling**- string constructors string length, charAt(), getChars(), equals(), concat(), replace(), trim(), StringBuffer		**11 Hrs**
Module II **Introducing Classes, Objects, and Methods**-Class Fundamentals, Declaring Objects , Assigning object Reference Variables , Methods, Constructors, Garbage Collection and Finalize method, The this Keyword **A Closer Look at Methods and Classes –** Overloading methods, Using objects as parameters, Argument passing, Returning objects**,** Access control , Understanding Static, Introducing Final, Nested and Inner classes, Using command line arguments		**10 Hrs**
Module III **Inheritance:** Inheritance Basics, Using super, Creating a Multilevel Hierarchy, When are Constructors are called, , Method Overriding, Using Abstact class,Using Final with inheritance, The Object Class,. **Interfaces:**Defining an interface,implementing interfaces,nested interfaces, applying interfaces, variables in interfaces, Interfaces can be extended, **Packages:** Packages, Access Protection , Importing Packages.		**10 Hrs**
Module IV **Exception Handling :** Exception Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch,Multiple catch clauses,Nested try statements,throw,throws,finally,java's built in exceptions,creating own exception subclasses **Multithreaded Programming** The java Thread Module,The Main Thread,Creating a Thread,Creating Multiple threads,thread priorities,Syncronization, Thread Communication using notify (), wait() and notify All(), suspending, Resuming and stopping Threads.		**10 Hrs**

Module V	**11 Hrs**
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. **JDBC-ODBC Connectivity:** JDBC program, using prepared Statement Object, Interactive SQL tool.	

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 Fundamentals: 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)
21CS44	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	Illustrate multithreading code for concurrency and develop GUI application program using Applet , event handling and database Connectivity.
	CO5	Design and develop web application using JDBC-ODBC connectivity.

Course Title: ANALYSIS AND DESIGN OF ALGORITHM LAB		
Subject Code : **21CSL45**	Credits : 1	CIE: 50
Number of Practical Hours/Week	**2 Hrs**	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 Selection sort and determine the time required to sort elements.

3. Write a C Program to sort a given set of elements using Merge sort method and determine the time required to sort the elements.

4. Write a C Program to Sort a given set of elements using Quick sort method and determine the time required sort the elements.

5. Write a C Program to Sort a given set of elements using Insertion sort and determine the time required to sort elements.

6. Write a C Program to Check whether a given graph is connected or not using DFS method.

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 Sort a given set of elements using the Heap sort method and determine the time required to sort the elements.

9. Write a C Program to Implement Horspool algorithm for String Matching.

10. Write a C Program to Implement Floyd‟s algorithm for the All-Pairs Shortest-paths.

11. Write a C Program to implement 0/1 Knapsack problem using dynamic programming problem.

12. Write a C Program to Find Minimum Cost Spanning Tree of a given undirected graph using Prim‟s algorithm. 13. Write a C Program to Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm. 14. 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. **15.** 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)
21CSL45	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 the graph 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: JAVA Lab		
Subject Code : **21CSL46**	Credit : 1	CIE: 50
Number of Practical Hours/Week	**2 Hrs**	SEE: 50
		SEE Hours: 03

Prerequisites: Concepts of Object Oriented Programming

Course Objectives:

- Learn to code and execute Java programs to solve problems
- Design of GUI for Java applications
- Understand 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 natural numbers 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 of strings. 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 Eclipse IDE)** 1. Write a Java Program to demonstrate the creation of class for student information. 2. Write a Java Program to implement inner class and demonstrate in access protections. 3. Write and execute a JAVA program to demonstrate use of any five string functions. Use both parameterized and non-parameterized constructors for passing string inputs. 4. Write and execute a JAVA Program to demonstrate Inheritance.(single level and multilevel) 5. Write and execute a JAVA Program to demonstrate exception handling (both built- in and user-defined exceptions). 6. Write and execute a JAVA Program to demonstrate polymorphism through method overloading . 7. Write and execute a JAVA program to demonstrate method overriding. 8. Write a Java program to implement multithreading in JAVA which demonstrate built in methods available for thread.	

9. Write a JAVA program which demonstrate, create and import packages in JAVA 10. Write a applet program and required HTML file to create banner applet. 11. Write a JAVA applet program to create a basic Applet having buttons, textarea GUI controls to add & subtract two nos. Use appropriate event listeners. 12. Write a Java program to store, delete and update data in a database with thesupport of JDBC-ODBC connectivity. **Open Ended Project : Servlets**	

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)
21CSL46	CO1	Implement Java programs with basic concepts of Object oriented programming.
	CO2	Demonstrate constructors ,Run-time and user-defined exceptions.
	CO3	Develop code for Inheritance, method overriding and overloading
	CO4	Design interactive GUI Java programs using applets and event handling programs
	CO5	Develop web application using JDBC-ODBC connectivity.

Course Title: WEB APPLICATION DEVELOPMENT LAB		
SubjectCode:**21CSL47**	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 HTML5 documents to study various HTML tags, style sheets and the tag, Borders, padding, color, and the tag.
2. Develop a JavaScript embedded HTML5 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. Run SQL queries to do the following: create a database, create table, insert rows in a table, fetch rows from a table, delete a row, and update a row.
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 intoitsMD5hash use table created in experiment

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

Questionpaperpattern:ForSEEsimilarquestionrelatedtotheaboveprogramswillbe asked.		
Course outcomes: **On completion of the course, the student will have the ability to:**		
Course Code	**CO#**	**Course Outcome(CO)**
21CSL47	**CO1**	Design of Static web programming usingHTML5.
	CO2	Create web pages using HTML5, Cascading Style Sheets, JavaScript.
	CO3	Design and implement dynamic Web pages with server side Information using Perl.
	CO4	Write PHP programs to for client server interaction.
	CO5	Develop database applications using MySQL database with PHP.

Course Title: SOFTWARE ENGINEERING AND TOOLS		
Subject Code : 2**1CS51**	Credits : 3	CIE: 50
Number of Lecture Hours/Week((L:T:P)	**3:0:0 Hrs**	SEE: 50
Total Number of Lecture Hours	42	SEE Hours: 03
Prerequisites: Any programming language		

Course objectives:

- Acquire knowledge of software development lifecycle
- 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 **Overview:** Introduction: FAQ's about software engineering, Professional and ethical responsibility. **Software Processes:** Software Processes: Models, Process iteration, Process activities, The Rational Unified Process, Computer-Aided Software Engineering. **Requirements:** Software Requirements: Functional and Non-functional requirements, User requirements, System requirements, Interface specification, and The software requirements document.	**08 Hrs**
Module - II **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 **Verification And Validation:** Verification and Validation: Planning, Software inspections, Automated static analysis, Verification and formal methods. **Management:** Managing People: Selecting staff, Motivating people, Managing people, The People Capability Maturity Model. Software Cost Estimation: Productivity, Estimation techniques.	**08 Hrs**
Module – IV **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	08 Hrs
Path Testing: DD paths, Test coverage metrics, Basis path testing, guidelines and observations. Define/Use testing, Slice-based testing, Guidelines and observations. **Levels of Testing**: Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing. Integration Testing: A closer look at the SATM system, Decomposition-based Integration, call graph-based Integration.	

10 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.
3. **Object Oriented System Development using UML**
   Ali Bahrami , MaGrawHill, 1999

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.

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

Course code	CO #	Course Outcome (CO)
21CS51	**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	Explain and 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: COMPUTER NETWORKS		
Subject Code: **21CS52**	Credit:4	CIE:50
Number of Lecture Hours/Week(L:T:P)	**3:0:2Hrs**	SEE:50
Total Number of Lecture Hours	42	SEE Hours: 03
Prerequisites :Nil		

Course Objectives:

- Develop an understanding about architectural principles of computer networks , network devices and their functions.
- Gain knowledge about functions and services of OSI layers and TCP/IP protocol.
- Learn how internet works, understand working of routing protocols and study implementation issues in internetworking.
- Understand transport and application layer protocols.

MODULES	**Teaching Hours**
Module I **Introductory concepts& Physical Layer:** Network Hardware, Network Software, Reference Models, Example Networks, The Theoretical Basis for Data Communication, Guided Transmission Media ,Wireless Transmission. 1. Experimental study of various network components and devices. a. Study different network cables and Prepare, test straight over and cross over cabling using crimping tool. b. Install and configure wired and wireless NIC. Demonstrate file transfer in wired and wireless LAN. c. Install and configure network devices hub. . Use CISCO packet tracer to . Build a Local Area Network of 4 to 6 nodes using hub /repeater. a. Build a peer to peer network	**08 Hrs**
Module II **Data Link Layer & Medium Access Control Sub-layer:** Data link layer design issues, Error detection & correction, Elementary data link protocols, Sliding window protocols, Example data link protocols**,** Th**e** channel allocation problem, Multiple access protocols. 1. Implement sliding window protocol. 2. Implement go back N protocol.	**08Hrs**

Module III **Medium Access Control Sub-layer:** Ethernet, Wireless LANS, Broadband Wireless, Bluetooth, Data link layer switching. 1. Install and configure network devices Switch. 2. Use CISCO packet tracer to a. Build a Local Area Network of 4 to 6 nodes using switch. b. Build a Local Area Network of 4 to 6 nodes using hub and a switch and study the differences between repeater, hub and switch. c. identify broadcast and collision domain. 3. Use wireshark to . Examine Ethernet packets and ARP packets. 4. To study performance of CSMA/ CD protocol.	**08 Hrs**
Module IV **The Network Layer:** Network layer design issues**,** Routing Algorithms, Congestion control algorithms, Internetworking, The network layer in the internet. 1.Install and configure network devices Routers. 2. Use CISCO packet tracer to a. Design and apply IP addressing scheme for a given topology b. Connect two or three LAN's via a router. Trace how routing happens via simulation, and study the working of router. c. Design multiple subnets with suitable number of hosts d. Demonstrate static routing and dynamic routing for given topology e. Configure DHCP server f. Create subnets , Configure Host IP, Subnet Mask and Default Gateway in a LAN g. Configure RIP/OSPF. h. Use wireshark toAnalyze IP Datagram and IP fragmentation received during the execution of trace route command. i. Run ping command and examine ICMP packets using wireshark.	**08 Hrs**
Module V **The Transport Layer and Application Layer protocols:** The transport services. Elements of transport protocols, The internet transport protocols: UDP The internet transport protocols: TCP, DNS-The Domain name system, Electronic mail, The world wide web. 1. Use wireshark to a. Examine UDP and TCP ports and handshake segments b. Use packet tracer to configure DHCP server, DNS server, SMTP server 2. Implement Client Server Program in C/ Java.	**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. Andrew S. Tanenbaum: Computer Networks, 5th Edition, Pearson,2010.
2. 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 with TCP/IP Protocol suite , Sixth Edition, McGraw Hill,2022.
2. Kurose and Ross, Computer Networking: A Top- Down Approach, Pearson, Sixth Edition,2021
3. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education,2007.
4. 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#	CourseOutcome(CO)
21CS52	CO1	Understand basic concepts, study OSI, TCP/IP model with functions of each layer and understand wired and wireless transmission fundamentals.
	CO2	Describe error detection, correction methods, data link layer functions and evaluate channel access mechanisms.
	CO3	Study and compare medium access protocols for wired and wireless LAN's
	CO4	Demonstrate routing layer functions, issues and routing protocols in Internet.
	CO5	Explore transport layer functions, issues and application layer protocols.

Course Title: OPERATING SYSTEM		
Subject Code:**21CS53**	Credit:3	CIE:50
Number of Lecture Hours/Week(L:T:P)	**3:0:0 Hrs**	SEE:50
Total Number of Lecture Hours	42	SEEHours:03
Prerequisites: 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 **Introduction:** Operating Systems, Computer-System Organization, Computer-System Architecture, Operating-System Operations, Process Management, Memory Management, Storage Management, Security and Protection, Kernel Data Structures, Computing Environments. **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. **Case Studies:** Architecture of UNIX, The Kernel of Unix; The Kernel of Solaris; Architecture of Windows.	**08 Hrs**
Module-II **Process Management**: Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication, Communication in Client–Server Systems. **Multithreaded Programming:** Overview, Multicore Programming, Multithreading Models, Thread Libraries, Implicit Threading, Threading Issues	**08 Hrs**
Module– III **Process Scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Thread Scheduling, Multi-Processor Scheduling, Real-Time CPU Scheduling **Process Synchronization:** The Critical-Section Problem, Petersons Solution, Synchronization hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors.	**09 Hrs**
Module– IV **Deadlocks:** System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock. **Memory Management:** Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table.	**08 Hrs**

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

Textbook:

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

Reference Books:

1. P.C.P. Bhatt: Operating Systems, 2nd Edition, PHI, 2006.
2. Harvey MDeital: 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)
21CS53	**CO1**	Describe the functions of operating systems and its structures
	CO2	Illustrate process concepts and management models.
	CO3	Apply Scheduling algorithms and differentconcurrencycontroltechniquestoprovideco-ordinationamongprocessesfortheglobaldata.
	CO4	Apply deadlock detection and prevention algorithms and memory management and illustrate the concept of paging, segmentation and swapping policies.
	CO5	Discuss Virtual memory management and describe file system interface.

Course Title: DATABASE MANAGEMENT SYSTEM		
Subject Code : **21CS54**	Credit :3	CIE: 50
Number of Lecture Hours/Week(L:T:P)	**3:0:0 Hrs**	SEE: 50
Total Number of Lecture Hours	42	SEE Hours: 03
Prerequisites: knowledge of C, C++ Programming Principles, Data Structures		
Course Objectives: • Learn and practice data modeling 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 **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.	**10 Hrs**
Module II **Relational Model:** Relational Model Concepts, Relational Model Constraints and Relational Database Schemas. The Relational Algebra and relational calculus, SQL-99: Schema Definition, Constraints, Queries, and Views, SQL Programming Techniques.	**8 hours**
Module III **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	**8 hours**
Module IV **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.	**8 hours**
Module V **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	**8 hours**

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	

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 Ramakrishnan and Johannes Gehrke – 3rd Edition. MCSraw-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)
21CS54	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: DATABASE MANAGEMENT SYSTEM LAB		
Subject Code : **21CSL55**	**Credits: 1**	CIE: 50
Number of Practical Hours/Week/batch (L:T:P)	**0:0:2 Hrs**	SEE: 50
		SEE Hours: 03

Prerequisite: 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:

1. Implementation of DDL commands of SQL with suitable examples.
    - Create table
    - Alter table
    - Drop Table
2. Implementation of DML commands of SQL with suitable examples
    - Insert
    - Update
    - Delete
3. Implementation of different types of function with suitable examples
    - Number function
    - Aggregate Function
    - Character Function
    - Conversion Function
    - Date Function
4. Implementation of different types of operators in SQL
    - Arithmetic Operators
    - Logical Operators
    - Comparison Operator
    - Special Operator
    - Set Operation
5. Implementation of different types of Joins
    - Inner Join
    - Outer Join
    - Natural Join etc..
6. Study and Implementation of
    - Group By & having clause
    - Order by clause
    - Indexing
7. Study & Implementation of
    - Sub queries

- Views

8. Study & Implementation of different types of constraints.
9. Study & Implementation of Database Backup & Recovery commands, Rollback, Commit, Savepoint.
10. Creating Database /Table Space, Managing Users: Create User, Delete User, Managing roles:-Grant, Revoke
11. Study & Implementation of PL/SQL.
12. Study & Implementation of SQL Triggers.

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

1. Inventory Control System.
2. Core Banking system
3. Hospital Management System.
4. Railway Reservation System.
5. Personal Information System.
6. Web Based User Identification System.
7. Timetable Management System.
8. Hotel Management System.
9. Library management
10. Electricity bill.
11. Hostel management.
12. Air reservation
13. Company management system.
14. Student information system.
15. 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 will be considered for CIE and SEE

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

Course Code	CO #	Course Outcome (CO)
21CSL55	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

RESEARCH METHODOLOGY & INTELLECTUAL PROPERTY RIGHTS

[As per Choice Based Credit System (CBCS) Scheme]
(From the academic year 2022-23)

Course Code	21RMI56	CIE Marks	50
Credits	02	SEE Marks	50
Course Type	Theory		
Lecture Hours/Week (L-T-P)	1-2-0-0	Total Marks	100
Total Hours	28 Hours	SEE Hours	03

Course Objectives:

CO1: To Understand the knowledge on basics of research and its types.
CO2: To Learn the concept of defining research problem and Literature Review,Technical Reading.
CO3: To learn the concept of attributions and citation and research design.
CO4: Concepts, classification, need for protection, International regime of IPRs - WIPO , TRIPS, Patent - Meaning, Types, surrender, revocation, restoration, Infringement , Procedure for obtaining Patent and Patent Agents.
CO5: Meaning, essential requirements, procedure for registration and Infringement of Industrial Designs, Copyright.

MODULES	Hours
Module-1 Introduction: Meaning of Research, Objectives of Engineering Research, and Motivation in Engineering Research, Types of Engineering Research, Finding and Solving a Worthwhile Problem. Ethics in Engineering Research, Ethics in Engineering Research Practice, Types of Research Misconduct, Ethical Issues Related to Authorship.	**06 Hours**
Module - 2 **Defining the research problem** - Selecting the problem. Necessity of defining the problem Techniques involved in defining the problem- Importance of literature review in defining a problem Literature Review and Technical Reading, New and Existing Knowledge, Analysis and Synthesis of Prior Art Bibliographic Databases, Web of Science, Google and Google Scholar, Effective Search: The Way Forward Introduction to Technical Reading Conceptualizing Research, Critical and Creative Reading, Taking Notes While Reading, Reading Mathematics and Algorithms, Reading a Datasheet.	**06 Hours**
Module - 3 **Research design** and methods - Research design - Basic principles. Need of research design Features of good design- Important concepts relating to research design - Observation and Facts **Attributions and Citations:** Giving Credit Wherever Due, Citations: Functions and Attributes, Impact of Title and Keywords on Citations, Knowledge Flow through Citation, Citing Datasets, Styles for Citations, Acknowledgments and Attributions, What Should Be Acknowledged, Acknowledgments in, Books Dissertations, Dedication or Acknowledgments.	**06 Hours**

Module - 4 Basic Concepts of Intellectual Property (IP), Classification of IP, Need for Protection of IP, International regime of IPRs - WIPO , TRIPS. **Patents:** Meaning of a Patent – Characteristics/ Features . Patentable and Non-Patentable Invention. Procedure for obtaining Patent. Surrender of Patent, revocation &restoration of Patents, Infringement of Patents and related remedies (penalties) . Different prescribed forms used in Patent Act. Patent agents-qualifications and disqualifications Case studies on patents - Case study of Neem petent, Curcuma(Turmeric)patent and Basmati rice patent, Apple inc.v Samsung electronics co.Ltd	**05 Hours**
Module - 5 **Industrial Design :** Introduction to Industrial Designs. Essential requirements of Registration. Designs which are not registrable, who is entitled to seek Registration, Procedure for Registration of Designs **Copy Right** Meaning of Copy Right. Characteristics of Copyright. Who is Author, various rights of owner of Copyright. Procedure for registration. Term of copyright, Infringement of Copyright and Its remedies. Software Copyright.	**05 Hours**

Assessment Details(both CIE and SEE)

The weight age of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE(Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

Three Unit Tests each of 20Marks(duration 01hour)

1. First test at the end of 5th week of the semester
2. Second test at the end of the 10th week of the semester
3. Third test at the end of the15th week of the semester

Two assignments each of 10Marks

4. First assignment at the end of 4th week of the semester
5. Second assignment at the end of 9th week of the semester Groupdiscussion/ Seminar/quizanyoneofthreesuitablyplannedtoattaintheCOsandPOsfor20 Marks (duration 01 hours)
6. At the end of the 13th week of the semester

The sum of three tests, two assignments, and quiz/seminar/group discussion will be out of 100 marks and will be **scaled down to 50marks** (to have less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods/question paper is designed to attain the different levels of Bloom's taxonomy as per the Outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (duration 03 hours)

1. The question paper will be set for 100marks.Marks scored shall be proportionally reduced to50 marks
2. The question paper will have ten questions. Each question is set for 20marks.
3. There will be 2questions from each module .Each of the two questions is under a module (with a maximum of 2 sub-questions).
4. The students have to answer 5 full questions, selecting one full question from each module.

Marksscoredbythestudentswillbeproportionallyscaleddownto50marks

Course Outcomes

At the end of the course the student will be able to:

CO1.To know them leaning of engineering research.
CO2.To know the defining of research problem and procedure of Literature Review.
CO3.To know the Attributions and Citations and research design.
CO4. Highlights the basic Concepts and types of IPRs and Patents
CO5.Analyse and verify the procedure for Registration of Industrial Designs & Copyrights.

Textbook

1. Research Methodology: Methods and Techniques C.R.Kothari, Gaurav Garg New Age International 4thEdition,2018
2. Dipankar Deb•RajeebDey,ValentinaE.Balas "EngineeringResearchMethodology",ISSN1868- 4394 ISSN 1868-4408 (electronic), Intelligent Systems Reference Library, ISBN 978-981-13- 2946-3 ISBN 978-981-13-2947-0 (eBook), https://doi.org/10.1007/978-981-13-2947-0.3
3. Dr. M.K. Bhandari"Law relating to Intellectual property" January 2017 (Publisher By Central Law Publications).
4. Dr. R Radha Krishna and Dr. S Balasubramanain "Text book of Intellectual Property Right". First edition, New Delhi 2008. Excel books.
5. P Narayan "Text book of Intellectual Property Right". 2017 ,Publisher: Eastern Law House

Reference Book:

1. DavidV.Thiel"ResearchMethodsforEngineers"CambridgeUniversityPress,978-1-107-03488- 4-
2. Nishith Desai Associates - Intellectual property law in India – Legal, Regulatory & Tax

NPTEL:

INTELLECTUAL PROPERTY by PROF.FEROZ ALI , Department of Humanities and Social Sciences IIT Madras

https://nptel.ac.in/content/syllabus_pdf/109106137.pdf

www.wipo.int

www.ipindia.nic.in

ENVIRONMENTAL STUDIES

[As per Choice Based Credit System (CBCS) Scheme]
(From the academic year 2022-23)

Course Code	21CIV57	CIE Marks	50
Credits	01	SEE Marks	50
Course Type	Theory		
Lecture Hours/Week (L-T-P)	0-2-0-0	Total Marks	100
Total Hours	28 Hours	SEE Hours	01

Course Objectives:

- To create environmental awareness among the students.
- To gain knowledge on different types of pollution in the environment.

Teaching-Learning Process(General Instructions)

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

1. Apart from conventional lecture methods various types of innovative teaching techniques through videos and animation films may be adopted so that the delivered lesson can progress the students in theoretical, applied and practical skills.
2. Environmental awareness program on off campus
3. Encourage collaborative (Group Learning) Learning in the class.Seminars, surprise tests and Quizzes may be arranged for students in respective subjects to develop skills.

Modules	Hours
Module - I Ecosystems (StructureandFunction): Forest, Desert, Wetlands, River, OceanicandLake. Biodiversity:Types,Value;Hot spots; Threatsand Conservation of biodiversity, Forest Wealth, And Deforestation.	05 Hours
Module - II Advances in Energy Systems (Merits, Demerits, Global Status and Applications): Hydrogen, Solar, OTEC, Tidal and Wind. Natural Resource Management (Concept and case-studies): Disaster Management, Sustainable Mining, case studying, and Carbon Trading	05 Hours
Module-III **Environmental Pollution** (Sources, Impacts, Corrective and Preventive measures, Relevant Environmental Acts, Case-studies): Surface and Ground Water Pollution; Noise pollution; Soil Pollution and Air Pollution. **Waste Management & Public Health Aspects** :Bio-medical Wastes; Solid waste; Hazardous, Wastes; E-wastes; Industrial and Municipal Sludge.	06 Hours

Module-IV **Global Environmental Concerns**(Concept, policies and case-studies): Groundwater depletion/recharging, Climate Change; Acid Rain; Ozone Depletion; Radon and Fluoride problem In drinking water; Resettlement and rehabilitation of people, Environmental Toxicology.	06 Hours
Module - V **Latest Developments in Environmental Pollution Mitigation Tools (Concept and Applications):** G.I.S. & Remote Sensing, Environment Impact Assessment, Environmental Management Systems, ISO14001; Environmental Stewardship- NGOs. Field work: Visit to an Environmental Engineering Laboratory or Green Building or Water Treatment Plant or Waste water treatment Plant; ought to be Followed by understanding of process and its brief Documentation in the form of report.	06 Hours

Course outcome(Course Skill Set)

At the end of the course the student will be able to:

CO1: Understand the principles of ecology and environmental issues that apply to air, land, and water issues on a global scale,

CO2: Develop critical thinking and/or observation skills, and apply them to the analysis of a problem or question related to the environment.

CO3: Demonstrate ecology knowledge of a complex relationship between biotic and a biotic components.

CO4: Apply their ecological knowledge to illustrate and graph a problem and describe the realities that managers face when dealing with complex issues.

CO5: Understand Latest Developments in Environmental Pollution Mitigation Tools Concept and Applications of G.I.S. & Remote Sensing.

Assessment Details (both CIE and SEE)

The weight age of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures not less than 35% (18 Marks out of 50)in the semester-end examination(SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE(ContinuousInternalEvaluation)andSEE(SemesterEndExamination) taken together.

Continuous Internal Evaluation:

Three Unit Tests each of **20Marks (duration 01 hour**)

1. First test at the end of5th week of the semester
2. Second test a the end of the10th week of the semester
3. Third test at the end of

the15thweek of the semester Two

assignments each of **10 Marks**

4. First assignment at the endof4thweek of the semester
5. Second assignment at the end of 9thweek of the semester

Group discussion/Seminar/quiz any one of three suitably planned to attain the Cos and Pos for **20 Marks** (**duration01hours**)

6. At the end of the13thweek of the semester

The sum of three tests, two assignments, and quiz/seminar/group discussion will be out of 100 marks and will be

Scaled down to 50 marks

(to have less tresses CIE, the portion of the syllabus should not be common/repeated for any of the method of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods/question paper is designed to attain the different levels of Bloom' s taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 01 hours)**

Question paper pattern:

1. The Question paper will have 50 objective questions.
2. Each question will before 01marks
3. Students will have to answer all the questions on an OMR Sheet.
4. The Duration of the Exam will be 01 hour

Suggested Learning Resources:

Books

1. Environmental studies, Benny Joseph, Tata Mcgraw -Hill 2nd edition 2012·
2. Environmental studies, SM Prakash , pristine publishing house, Mangalore3rd edition-2018.

Reference Books:-·

1. BennyJoseph, Environmental studies,TataMcgraw-Hill2nd edition 2009·
2. M. Ayi Reddy TextbookofenvironmentalscienceandTechnology,BSpublications2007·
3. Dr.B.SChauhan,Environmentalstudies,universityofsciencepress1st edition

Course Title: PYTHON PROGRAMMING		
Subject Code : **21CSAE581**	Credit : 1	CIE: 50
Number of Practical Hours/Week(L:T:P)	**0:0:2 Hrs**	SEE: 50
Total Number of Practical Hours	**24**	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 to solve scientific problems.
- Use Python lists, tuples, sets and dictionaries for representing compound data.
- Develop structured Python programs by defining functions and calling them.
- Develop object oriented programming concepts in Python.
- Basic data analysis and visualization by Numpy and matplotlib libraries.

Programs

1. The structure of Python Programming through example programs.
2. Demonstrate the working of all kinds of operators.
3. Demonstrate the decision making and Iterative statements in Python
   i) "if" and its variants ii) while and for loops.
4. Demonstrate the use of various string functions like count (), replace (), split (), join (), upper (), lower (), capitalize () etc.
5. Demonstrate the file operations in python.
6. Demonstrate creation and different operations on List data structure in python.
7. Demonstrate creation and different operations on Tuple data structure in python.
8. Demonstrate creation and different operations on Set data structure in python.
9. Demonstrate creation and different operations on Dictionary data structure in python.
10. Demonstrate creation and use of Functions in python with all kinds of "parameters" used with functions.
11. Demonstrate different sorting operations in python and complex time difference.
12. Demonstrate creating objects and inheritance.
13. Demonstrate NumPY library – Array Operations, Mathematical Functions, Sort, Search and Counting Functions.
14. Demonstrate Matplotlib Library – Introduction, Pyplot API, Types Of Plots, Histogram Using Matplotlib , I/O With Numpy.

Text books:

1. Learning Python, Mark Lutz, Orielly, 3rd Edition 2007.
2. Think Python, 2nd Edition, 2017 Allen Downey, Green Tea Press

Reference Links: https://www.w3schools.com/python/
https://www.geeksforgeeks.org/python-programming-language/

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

Course Code	CO #	Course Outcome (CO)
21CSAE581	**CO1**	Understand python structure and use of operators, string functions, conditional and looping statements.
	CO2	Use of Python lists, tuples, sets and dictionaries for representing compound data.
	CO3	Develop modular python programs by defining functions.
	CO4	Implement programs with object oriented concepts.
	CO5	Develop program to utilize Numpy libraries for data analysis and visualize data with matplotlib library.

Course Title: ENTREPRENEURSHIP, MANAGEMENT AND FINANCE		
Subject Code : 2**1HU61**	Credits : 3	CIE: 50
Number of Lecture Hours/Week(L:T:P)	**3:0:0 Hrs**	SEE: 50
Total Number of Lecture Hours	42	SEE Hours: 03
Prerequisites: None		

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 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	**08 Hrs**
Module - II MANAGEMENT: Introduction – Meaning – nature and characteristics of Management, Scope and functional areas of management, Levels of Management, Henry Fayol - 14 Principles to Management , McKinsey's 7-S Model, Management by objective(MBO) – Meaning, process of MBO, benefits and drawbacks of MBO	**09 Hrs**
Module - III PREPARATION OF PROJECT AND SOURCE OF FINANCE: PREPARATION OF PROJECT: Meaning of project; Project Identification; Project Selection; Project Report; Need and Significance of Report; Contents; 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)	**08 Hrs**
Module – IV 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	**09 Hrs**
Module – V PERSONNEL MANAGEMENT, MATERIAL MANAGEMENT AND INVENTORY CONTROL: PERSONNEL MANAGEMENT: Functions of Personnel Management, Recruitment, Selection and Training, Wages, Salary and Incentives. MATERIAL MANAGEMENT AND INVENTORY CONTROL: Meaning, Scope and Objects of Material Management. Inventory	08 Hrs

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)	

Pattern of question paper

Solve all five full questions selecting at least one question from each module

Text book:

1. Financial Accounting -B S RAMAN- United Publishers Manglore, Maheswar S N & Maheswari S K- Vikas Publishing House. January 2018
2. Management & Entrepreneurship- K R Phaneesh- Sudha Publications January 2018 ,Prof Manjunatha & Amit kumar G – laxmi Publication , January 2011. Veerbhadrappa Havina -Published by New Age International (P) Ltd., 2009.
3. Principles of Management First Edition (English, G. Murugesan), Laxmi Publications – New Delhi
4. Management by Objectives (Mbo) in Enterprises: 21 December 2018 by Dr Wazir Ali Khan

Reference Books:

1. Industrial Organization & Engineering Economics-T R Banga & S C Sharma- Khanna Publishers, Dehli.
2. NPTEL : ENTREPRENEURSHIP: PROF. C BHAKTAVATSALA RAO Department of Management Studies IIT Madras https://nptel.ac.in/courses/110/106/110106141/
3. https://www.businessmanagementideas.com/notes/management-notes/notes-on-management-in-an-organisation/4669
4. https://vskub.ac.in/wp-content/uploads/2020/04/Unit-5-ppmb.pdf

Course outcomes:

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

Course code	CO #	Course Outcome (CO)
21HU61	CO1	Develop Entrepreneurship skills
	CO2	Apply the concepts of management and Management By Objective(MBO)
	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 GRAPHICS AND FUNDAMENTALS OF IMAGE PROCESSING		
Subject Code : 21CS62	Credits :03	CIE: 50
Number of Lecture Hours/Week(L:T:P)	**3:0:2 Hrs**	SEE: 50
Total Number of Lecture Hours	42 Hrs	SEE Hours: 03

Prerequisites: Nil

Course Objectives:

- Identity and explain the core concepts of computer graphics.
- Apply graphics programming techniques and create effective OpenGL programs.
- To Study the Image fundamental and mathematical transformations necessary for image processing.
- Understand the image enhancement techniques, image restoration and segmentation techniques.

MODULES	**Teaching Hours**
Module - I **Basics of Computer Graphics and OpenGL:** Computer Graphics: Basics of computer graphics, Application of Computer Graphics, Video Display Devices, graphics software. OpenGL: Introduction to OpenGL ,coordinate reference frames, specifying two-dimensional world coordinate reference frames in OpenGL, OpenGL point functions, OpenGL line functions, point attributes, line attributes, curve attributes, OpenGL point attribute functions, OpenGL line attribute functions, Line drawing algorithms(DDA, Bresenham's), circle generation algorithms (Bresenham's).	**08 Hrs**
Module - II **Fill area Primitives, 2D Geometric Transformations and 2D viewing :** Fill area Primitives: Polygon fill-areas, OpenGL polygon fill area functions, fill area attributes, general scan line polygon fill algorithm, OpenGL fill-area attribute functions. 2DGeometric Transformations: Basic 2D Geometric Transformations, matrix representations and homogeneous coordinates. Inverse transformations, 2DComposite transformations, other 2D transformations, raster methods for geometric transformations, OpenGL raster transformations, OpenGL geometric transformations function, 2D viewing: 2D viewing pipeline, OpenGL 2D viewing functions.	**08 Hrs**
Module – III **Digital Image Fundamentals:** Introduction to Digital Image Processing, Examples of fields that use DIP, Fundamental Steps in Digital Image Processing, Image Sensing and Acquisition: image acquisition using a single sensing element, image acquisition using sensor strips, image acquisition using sensor arrays, a simple image formation model, Image Sampling and Quantization: basic concepts in sampling and quantization, representing digital images, Some Basic Relationships between Pixels.	**08 Hrs**
Module – IV **Image Enhancement in the Spatial Domain:** Basics of intensity transformations and spatial filtering, Some Basic Intensity Transformation Functions, Histogram Processing: Histogram equalization, and Matching, Fundamentals of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, **Frequency Domain:** Preliminary Concepts, The Discrete Fourier Transform (DFT) of One variable, The Discrete Fourier Transform (DFT) of Two Variables.	**09Hrs**

Module – V	
Restoration: A model of the image degradation/restoration process , Noise models, Restoration in the Presence of Noise Only using Spatial Filtering and Frequency Domain Filtering **Image segmentation**: Detection of discontinuities, edge linking and boundary detection, Thresholding, Region based segmentation	**09Hrs**

Lab Programs:

1. Program to draw points , line, circle, Polygon and rectangle on a plane using OpenGL.
2. Program to draw a color cube and spin it using OpenGL transformation matrices
3. Rotation of House about Fixed Point
4. Program to fill any given polygon using scan – line area filling algorithm
5. Program to draw a rotating cube with texture.
6. Program to demonstrate DDA Line Drawing Algorithm.
7. a).Program to demonstrate Bresenham's Line Drawing Algorithm
   b). Program to demonstrate Bresenham's Circle Drawing Algorithm
8. a). Simulation and Display of an Image, Negative of an Image(Binary & Gray Scale)
   b). Implementation of Relationships between Pixels
9. Contrast stretching of a low contrast image, Histogram, and Histogram Equalization
10. Computation of Mean, Standard Deviation, Correlation coefficient of the given Image
11. Implementation of Image Smoothening Filters(Mean and Median filtering of an Image).
12. Perform noise removal using different spatial filters and compare their performances.
13. Perform the following Image segmentation operations: Edge detection, line detection and point detection.
14. Implement region based segmentation of image.

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. Donald Hearn & Pauline Baker: Computer Graphics with OpenGL Version,3rd / 4th Edition, Pearson Education,2011
2. ***Gonzalez*** and. Richard E. Woods' ***Digital Image Processing***, Fourth Edition, Global Edition 2018.

Reference Books:

1. Digital Image Processing- S.Jayaraman, S. Esakkirajan,T. Veerakumar, TataMcGrawHill2014.
2. Digital Image Processing (with Matlab and Labview), Vipul singh, elsiver.Filip learning
3. William M Newman and Robert F Sproull, Principles of Interactive Computer Graphics, McGraw Hill, 2001.

Course outcomes:

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

Course Code	CO #	Course Outcome (CO)
21CS62	**CO1**	Describe the basics of Computer Graphics and OpenGL.
	CO2	Illustrate 2D Transformations and Viewing.
	CO3	Describe the fundamentals concepts of digital image processing
	CO4	Demonstrate the techniques for Image enhancement in Spatial and frequency domain.
	CO5	Analyze Images restoration and Segmentation operations.

Course Title: ARTIFICIAL INTELLIGENCEAND MACHINE LEARNING		
Subject Code : **21CS63**	Credit :03	CIE: 50
Number of Lecture Hours/Week(L:T:P)	**3:0:0 Hrs**	SEE: 50
Total Number of Lecture Hours	42	SEE Hours: 03

Prerequisites: Discrete Mathematics, Statistics.

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.
- Acquiring the fundamentals of machine learning
- Usage of various learning methods to develop an intelligent machine.

MODULES	Teaching Hours
Module I **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. **Heuristic search techniques:** Generate-and-test, Hill climbing, Best-first search, Problem reduction.	**09 Hrs**
Module II **Knowledge representation issues:** Representations and mappings, Approaches to knowledge representation, Issues in knowledge representation, the frame problem.**Using predicate logic:** Representing simple facts in logic, representing instance and ISA relationships, Computable functions and predicates, Resolution, Natural Deduction **Representing Knowledge Using Rules:** Procedural versus Declarative knowledge, Logic programming, forward versus backward reasoning, matching, control knowledge.	**08 Hrs**
Module III Well posed learning problems, Designing a Learning system, Perspective and Issues in Machine Learning.**Concept Learning:** Concept learning task, Concept learning as search, Find-S algorithm, Version space, Candidate Elimination algorithm, Inductive Bias.	**08 Hrs**

Module – IV **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. **Artificial Neural Networks:** Introduction, Neural Network representation, Appropriate problems, Perceptrons, Multilayer networks and the Back propagation algorithm.	**09 Hrs**
Module V **Instance Based Learning:** Introduction, k-nearest neighbour learning, locally weighted regression, radial basis function, cased-based reasoning. **Bayesian Learning:** Introduction, Bayes 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.	**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. Tom M. Mitchell, "***Machine Learning",*** Indian Edition Paperback 2017, McGraw Hill Education.

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.

3. Trevor ***"The Elements of Statistical Learning"***, 2nd edition, 2017, Springer series in statistics. Hastie, Robert Tibshirani, Jerome Friedman

4. Ethem Alpaydın, "***Introduction to machine learning"***, Third Edition, PHI Learning Pvt. Ltd. 2015

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

Course Code	CO #	Course Outcome (CO)
21CS63	**CO1**	Discuss artificial intelligence techniques, problem and heuristic search algorithm
	CO2	Apply knowledge representation techniques and predicate Logic rules to solve reasoning programs.
	CO3	Identify the problems for machine learning.
	CO4	Apply supervised/ unsupervised learning for the given problem and Explain theory of probability and statistics related to machine learning.

	CO5	Estimate target function using Instance based learning

COURSE TITLE: COMPILER DESIGN AND SYSTEM SOFTWARE		
Subject Code : 21CS641	Credits :3	CIE: 50
Number of Lecture Hours/Week (L:T:P)	**3:0:0 Hrs**	SEE: 50
Total Number of Lecture Hours	42	SEE Hours: 03
Prerequisite : Finite Automata and Formal Languages.		
Course Objectives : • Understand the Process involved in constructing compilers. • Understand various types of parsers, intermediate code generation, Target code generation, Optimization of target code.		

Modules	**Teaching Hours**
Module I **Introduction**: Language Processors , The Structure of a Compiler, The Science of Building a Compiler, Applications of Compiler Technology. **Simple Syntax directed Translator:** Syntax Definition, Syntax Directed Translation, A translator for simple Expressions, Symbol Tables , Intermediate code generation. **Lexical Analysis**: the Role of Lexical Analyzer, Input buffering, specification of tokens, reorganization of tokens, the lexical analyzer generator Lex .	**08 Hrs**
Module II **Syntax Analysis:** Introduction to Recursive-Descent, Top-Down parsing, Bottom-Up parsing, LL(1),Shift/Reduce , Operator Precedence, LR(0), SLR(1), LR(1), SLAR(1) and LALR(1) parsers, Parser generators-Yacc.	**08 Hrs**
Module III **Syntax Directed Translation:** Syntax directed definitions, Evaluation orders for SDDs, Applications of syntax directed translation, Syntax directed Translations schemes. **Intermediate code generation:** Variants of syntax trees, three address code, pipes and declarations, translations of expression, Type checking, Control flow, Back patching, Switch statements, Intermediate code for processors.	**09 Hrs**
Module IV **Code Generation :** Issues in the design of code generator, The target language, Address in the target code, Basic blocks and flow graphs, Optimization of basic blocks, A simple code generator, Peephole optimization, register allocation and assignment, Instructions selection by tree rewriting, Optimal code generation for expressions.	**08 Hrs**
Module V **Assemblers:** Basic Assembler Functions, Machine-Dependent Assembler Features, Machine-Independent Assembler Features, Assembler Design Options, **Loaders and Linkers:** Basic Loader Functions, Machine- Dependent Loaders Features, Machine-Independent Leader Features, Loader Design Option.	**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. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman: Compilers - Principles, Techniques and Tools, 2nd Edition, Pearson, 2007.
2. Leland L. Beck, D.Manjula : System Software "An Introduction to System Programming", 3rd Edition 2008

Reference Books:

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

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

Course Code	CO #	Course Outcome (CO)
21CS641	CO1	Describe the Science of Building a Compiler, Specification and recognition of Tokens using Lexical Analyzer tool – Lex.
	CO2	Design and an analysis of Top-Down, Bottom-up, LR, LALR parsers and usage of Yacc tool to build parsers.
	CO3	Understanding SDD, SDT schemes and describe techniques for intermediate code generation.
	CO4	Demonstrate techniques for simple and optimal machine code generators.
	CO5	Understanding basic functions of assemblers, Loaders and Linkers.

Course Title: DESIGN OF IOT SYSTEM		
Subject Code : **21CS642**	Credits :03	CIE: 50
Number of Lecture Hours/Week(L:T:P)	**3:0:0 Hrs**	SEE: 50
Total Number of Lecture Hours	42 Hrs	SEE Hours: 03

Prerequisites: Microprocessors and Microcontrollers

Course Objectives:

- Understand basics of embedded systems and their design concepts
- Introduce IoT technology and its communication mechanisms
- Understand programming IoT development boards like Arduino and Raspberry pi
- Acquire the data with sensors and perform data analysis

MODULES	**Teaching Hours**
Module I Introduction to Embedded Systems, Processor Embedded into a System, Embedded Hardware Units and Devices in a System, Embedded Software in a System ,Examples of Embedded Systems ,Embedded System-on-chip (So) and Use of VLSI Circuit Design Technology, Complex Systems Design and Processors, Design Process in Embedded System, Formalization of System Design, Design Process and Design Examples, Classification of Embedded Systems, Skills required for an Embedded system designer.	**08 Hrs**
Module II 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.	**08 Hrs**
Module III Smart Objects: The "Things" inIoT, Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects, Communications Criteria, IoT Access Technologies: IEEE802.15.4, IEE802.15.4g, IEE802.15.4e and 19012a, IP as the IoT Network Layer, The Business Case for IP, The need for Optimization, Optimizing IP from 6LowPAN to 7Lo. Application Layer Protocols: Generic Web based protocols, COAP, MQTT protocol.	**08 Hrs**
Module IV Data and Analytic s for IoT, An Introduction to Data Analytics for IoT, Machine Learning, Big Data Analytic Tools and Technology, Edge Streaming Analytic, Network Analytics, Securing IoT. Introduction to Arduino, Arduino UNO, Installing the Software, Fundamentals of Arduino Programming.	**08 Hrs**
Module V Raspberry Pi: Introduction to Raspberry Pi, About the Raspberry Pi Board: Hardware Layout, Operating Systems on Raspberry Pi, Configuring Raspberry Pi, Programming	**08 Hrs**

Raspberry Pi with Python, Wireless Temperature Monitoring System Using Pi, DS18B20 Temperature Sensor, Storing data into remote data server.	

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. Rajkamal, "Embedded System Architecture, Programming and Design", second edition Tata McGraw- Hill publishing company limited.
2. 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.
3. Srinivasa K G, "Internet of Things", CENGAGE LeaningIndia,2017
4. Internet Of Things A hands on Approach, Arashdeep Bhaga, Vijay Madiseeti

Reference Books:

1. Vijay Madisetti and Arshdeep Bahga, "Internet of Things (AHands-on-Approach)", 1st Edition, VPT, 2014.
2. RajKamal, "Internet of Things: Architecture and Design Principles", 1st Edition, McGraw Hill Education, 2017.

Course outcomes:

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

Course Code	CO #	Course Outcome (CO)
21CS642	**CO1**	Understanding embedded system and its classification.
	CO2	Illustrate the impact and challenges posed by IoT networks leading to new architectural models.
	CO3	Deployment of smart objects and the technologies to connect them to network and its protocols for efficient network communication.
	CO4	Describe the need for Data analytics and Security in IoT. Understand Arduino board and programming and developing simple projects using Arduino UNO board.
	CO5	Understand Raspberry pi board and programming and develop simple projects using Raspberry pi and sensors.

COURSE TITLE: CRYPTOGRAPHY AND INFORMATION SECURITY		
Subject Code : 21CS643	Credits :03	CIE: 50
Number of Lecture Hours/Week (L:T:P)	**3:0:0 Hrs**	SEE: 50
Total Number of Lecture Hours	42 Hrs	SEE Hours: 03

Prerequisites: Mathematics.

Course Objectives:

- To Gain knowledge of secure network architecture
- Explain the mathematics and theory behind different cryptographic algorithms.

MODULES	Teaching Hours
Module - I **Introduction:** Security goals, Attacks, Services and Mechanism, Techniques. **Mathematics of Cryptography:** Integer arithmetic, Modular arithmetic, Linear congruence. **Traditional Symmetric Key Ciphers:** Introduction, Substitution Ciphers, Transposition Ciphers, Stream and Block Ciphers	**09 Hrs**
Module - II **Mathematics of Cryptography:** Algebraic structures, GF(2^n) Fields. **Introduction to modern Symmetric-Key Ciphers:** Modern Block Ciphers, Modern Stream Ciphers. **Data Encryption Standard(DES):**Introduction, DES Structure, DES Analysis, Multiple DES, Security of DES	**09 Hrs**
Module – III **Advanced Encryption Standard:** Introduction, Transformations, Key Expansion, Ciphers, Examples, Analysis of AES. **Encipherment Using Modern Symmetric-Key Ciphers:** Use of Modern Block Ciphers, Use of Stream Ciphers, Other issues. **Mathematics of Asymmetric key Cryptography:** Primes, Primality Testing, Factorization, Chinese Remainder Theorem, Quadratic Congruence, Exponentiation and Logarithm	**08 Hrs**
Module – IV **Asymmetric-Key Cryptography:** Introduction, RSA Cryptosystem, Rabin Cryptosystem, Elliptic Curve Cryptosystem. **Message Integrity and Message Authentication:** Message Integrity, Random Oracle Model, Message Authentication. **Cryptographic Hash Functions:** Introduction, SHA-512, Whirlpool	**08 Hrs**
Module – V **Digital Signature:** Comparison, Process, Services, Attacks on Digital Signature, Digital Signature Schemes, Variations and Applications. **Entity Authentication:** Introduction, Passwords, Challenge-Response, Zero-Knowledge, Biometrics. **Key Management:** Symmetric-Key distribution, Kerberos, Symmetric-Key Agreement, Public-Key Distribution	**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.Forouzan,B.A.―CryptographyandNetworkSecurity‖,TataMcGraw-Hill,2007

References

1. William Stallings, "Cryptography and Network Security", Pearson Education,2006
2. Atul Kahate ―Cryptography and Network Security, Tata McGraw-Hill, 2008

Course outcomes:

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

Course Code	CO #	Course Outcome (CO)
22CS643	CO1	Describe basic concepts of Cryptography and information security
	CO2	Apply algebraic structures to design encryption algorithms.
	CO3	Demonstrate AES algorithms and illustrate mathematical concepts behind design of asymmetric key cryptography and encipherment algorithms
	CO4	Discuss various algorithms for asymmetric key cryptography and message authentication
	CO5	Explain digital signatures and entity authentication

Course Title: Introduction to Artificial Intelligence		
Subject Code : **21CS65OE1**	Credit :3	CIE: 50
Number of Lecture Hours/Week(L:T:P)	**3:0:0 Hrs**	SEE: 50
Total Number of Lecture Hours	42	SEE Hours: 03
Prerequisites: Nil		

Course Objectives: This course will enable students to

- Identify the problems where AI is required and the different methods available
- Compare and contrast different AI techniques available.
- Know the applications of artificial Intelligence.
- Define and explain learning algorithms.

MODULES	**Teaching Hours**
Module I **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	**09Hrs**
Module II **Heuristic search techniques:** Generate-and-test, Hill climbing, Best-first search, Problem reduction, Mean-ends analysis. **Knowledge representation issues:** Representations and mappings, Approaches to knowledge representation, Issues in knowledge representation, the frame problem.	**09 hours**
Module III **Using predicate logic:** Representing simple facts in logic, representing instance and ISA relationships, Computable functions and predicates, Resolution, Natural Deduction **Representing Knowledge Using Rules:** Procedural versus Declarative knowledge, Logic programming, forward versus backward reasoning, matching, control knowledge.	**08 hours**
Module IV **Learning, Expert Systems :**Expert System, Knowledge representation ,Expert System shells, Knowledge Acquisition af an expert system, application of expert systems, Example of expert system, **Learning:** Rote Learning, Learning by Taking Advice, Learning in Problem Solving, Learning from Examples, Winston's Learning Program, Decision Trees	**08 hours**
Module V **Logic in Artificial Intelligence:** Proposition Logic, First Order logic **Prolog:** Logic programming symbolic logic, clausal form, converting English to prolog facts and rules, prolog terminology, variables and arithmetic operators, inference process of prolog, tracking model of execution, list structures, operations on list.	**08 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. Applications and Concepts , Techniques and Applications of Artificial Intelligence, Shirai, Yoshiaki and jun-ichi Tsujji, Published by John Wiley & Sons, Chichester, England, 1984,

Reference Books:

1. Artificial Intelligence: A Modern Approach, Stuart Rusell, Peter Norving, Pearson Education 2nd Edition.
2. Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems – Prentice Hal of India.
3. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problem Solving", Fourth Edition, Pearson Education, 2002.
4. Artificial Intelligence and Expert Systems Development by D W Rolston-Mc Graw hill.
   N.P. Padhy "Artificial Intelligence and Intelligent Systems" , Oxford University Press-2015

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

Course Code	CO #	Course Outcome (CO)
21CS65OE1	CO1	Identify the AI based problems
	CO2	Apply techniques to solve the AI problems
	CO3	Define learning and explain various learning techniques
	CO4	Discuss on expert systems
	CO5	Discuss on Logic in Artificial Intelligence

Course Title: ARTIFICIAL INTELLIGENCEAND MACHINE LEARNING LAB		
Subject Code : **21CSL66**	Credit :01	CIE: 50
Number of Practical Hours/Week/batch(L:T:P)	**0:0:2 Hrs**	SEE: 50
	SEE Hours: 03	

Prerequisites: Discrete Mathematics , Statistics, Java/Python Programming

Course Objectives:

Learn implementation and applications of Artificial Intelligence Algorithms.
Learn implementation and applications of Machine Learning Algorithms.
Understand the usage of various datasets for implementing ML Algorithms.

PROGRAMS

1. Write a Program to Implement Tic-Tac-Toe game using Python.
2. Write a Program to implement 8-Puzzle problem using Python.
3. Write a Program to Implement Water-Jug problem using Python.
4. Write a Program to Implement AO* Algorithm using Python.
5. 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.
6. 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.
7. Write a program to demonstrate the working of the decision tree basedID3 algorithm.
8. Use an appropriate data set for building the decision tree and apply this knowledge to classify anew sample. Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate datasets.
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
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.
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)
21CSL66	**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 :**21CSMP67**	**Credit : 2**	CIE: 50
Number of Practical Hours/Week(L:T:P)	**0:0:2 Hrs**	
Pre-requisite: Programming languages, Operating Systems		
Course Objectives: • Acquire the ability to integrate different areas of knowledge and evaluate and formulate aproblem • Acquire skills to communicate effectively and present their ideas and collaborate to work asa 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 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)
21CSMP67	**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 cyclemodel for project implementation and well-organized report