Engineering Mathematics-IV (Common to all branches)

Contact Hours/ Week	3(L)	**Credits**	3
Total Lecture Hours	39	**CIE Marks**	50
Total Tutorial Hours	00	**SEE Marks**	100
Sub Code	20MA410	**Semester**	IV

Prerequisites: Engineering Mathematics-I and II.

Course objectives:

1	Learn the concepts of finite differences, interpolation and their applications.
2	Evaluate Fourier transforms and use Z-transform to solve difference equations
3	Understand the concepts of calculus of functions of complex variables.
4	To make the student to solve system of linear equations, carryout matrix operations, determine the eigenvalues & eigenvectors.

Unit No.	Course Content	No. of Hours
UNIT-I	**Numerical Differentiation and Numerical Integration**: Derivatives using Newton-Gregory forward and backward interpolation formulae, Newton-Cotes quadrature formula, Trapezoidal rule, Simpson's1/3rd rule and Simpson's 3/8th rule. Applications to Engineering problems**.**	7 Hours
UNIT-II	**Fourier Transforms:** Finite and Infinite Fourier transform, basic properties, convolution theorem; inverse transforms **Z-transforms:** z-transforms and inverse z-transforms; solution of difference equation	8 Hours
UNIT-III	**Complex Variables-I:** Functions of complex variables, Analytic function, Cauchy-Riemann equations in Cartesian and polar coordinates, Consequences of Cauchy-Riemann equations, Construction of analytic functions	8 Hours
UNIT-IV	**Complex Variables-II:** Complex integration, Cauchy theorem, Cauchy integral formula, Taylor and Laurent series (statements only), Singularities, Poles and residues, Cauchy residue theorem.	8 Hours
UNIT-V	**Linear Algebra:** System of linear equations, Row operations, Echelon form Reduced Echelon form, Solution of Homogeneous and Nonhomogeneous equations, vector equations, Linear combinations, Linear independent/dependent vectors, Eigen values, Eigen vectors, Diagonalizations, solving a system of differential equations using diagonalization	8Hours

Text Books:

Sl. No.	Author/s	Title	Publisher Details
1	B.S.Grewal	"Higher Engineering Mathematics", 43rd edition	Khanna Publications, 2015.
2	Ramana B.V	Higher Engineering Mathematics", latest edition	Tata-McGraw Hill, 2016
3	Ralph P. Grimaldi	Discrete and Combinatorial Mathematics, 4th Edition,	PHI/Pearson Education, 2005
4	Erwin Kreyzig	, "Advanced Engineering Mathematics", 10th edition,	Wiley Publications, 2016

Reference Books:

Sl. No.	Author/s	Title	Publisher Details
1	Erwin Kreyszig	"Advanced Engineering Mathematics", 10th edition	Wiley Publications, 2015.
2	C. Ray Wylie and Louis C. Barrett	Advanced Engineering Mathematics", 6th edition,	Tata-McGraw Hill 2005.
3	Louis A. Pipes and Lawrence R. Harvill,	Applied Mathematics for Engineers and Physicists", 3rd edition,	McGraw Hill 2014.
4	Ralph P. Grimaldi	Discrete and Combinatorial Mathematics, 4th Edition	PHI/Pearson Education, 2005.
5	Maurice D. Weir, Joel R. Hass and George B. Thomas,	Thomas' calculus: Early Transcendentals", 12th edition,	Pearson Education, 2016
6	Ramana. B.V	"Higher Engineering Mathematics", 11th edition,	Tata-McGraw Hill. 2010
7	C.Ray Wylie, Louis. C. Barrett	Advanced Engineering Mathematics", 6th edition	Tata-McGraw Hill 2005.
8	Louis A. Pipes and Lawrence R. Harvill	Applied Mathematics for Engineers and Physicists", 3rd edition	McGraw Hill, 2014

Course Outcomes:

Upon completion of this course the student will be able to:

CO-1	Learn the concepts of finite differences, interpolation and their applications.
CO-2	Apply numerical techniques to solve Engineering problems and fit a least squares curve to the given data
CO-3	Evaluate Fourier transforms and use Z-transform to solve difference equations

CO-4	Examine and construct the analytic functions
CO-5	Classify singularities of complex functions and evaluate complex integrals.
CO-6	Ability to solve system of linear equations, carryout matrix operations, determines the eigenvalues & eigenvectors.

Mapping Course Outcomes with Program outcomes & Program Specific outcomes

Course Outcomes	Program Outcomes												PSO's			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	2	2											2			
CO2	2	2											2			
CO3	2	2											2			
CO4	2	2											2			
CO5	2	2											2			

Course Title: Operating Systems	***Course Code: 20CS410***
Credits (L: T:P): 4:0:0	***Contact Hours (L: T: P): 52:0:0***
Type of Course: Lecture	***Category: Professional Core Course***
CIE Marks: 50	***SEE Marks: 100***

Pre-Requisites: Computer Organization and Architecture, Data Structures.

Course objectives:

1	To understand the concepts and terminology used in OS.
2	Introduce threading and multithreaded systems.
3	Illustrate process synchronization and concept of Deadlock.
4	Introduce Memory and Virtual memory management, File system and storage techniques.

Unit No.	Course Content	No. of Hours
1	**Introduction to Operating Systems and System structures:** What operating systems do; Computer System organization; Computer System architecture; Operating System structure; OS operations; Process management; Memory management; Storage management; Protection and security; Distributed system; Special-purpose systems; Computing environments, services System structure: User - Operating System interface; System calls and its types System programs; OS design and implementation; OS structure; Virtual machines; OS generation; System boot.	10
2	**Process Management:** Process concept; Process scheduling; Operations on processes; Inter-process communication. Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms;	11
3	**Process Synchronizationand Deadlocks:** Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors. Deadlocks: System model; Deadlock characterization; Methods for	11

	handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.	
4	**Memory Management:** Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.	10
5	**File System, Secondary Storage Structures and Protection:** File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management. Secondary storage Structure: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights	10

Text Books:

Sl. No.	Author/s	Title	Publisher Details
1	Abraham Silberschatz, Peter Baer Galvin, Greg Gagne	Operating System Principles	9th Edition, Wiley India, 2013

Reference Books:

Sl. No.	Author/s	Title	Publisher Details
1	D.M Dhamdhere	Operating systems - A concept-based Approach	4th Edition, Tata McGraw-Hill, 2013
2	P.C.P. Bhatt	Introduction to Operating Systems	Concepts and Practice, 4th Edition, PHI, 2014
3	William Stallings	Operating Systems: Internals and Design Principles	7th Edition,Prentice Hall of India,2017
4	Harvey M Deital	Operating systems	3rd Edition, Pearson Education,2007

Web Resources:

Sl. No.	Web link
1	https://www.youtube.com/playlist?list=PLLDC70psjvq5hIT0kfr1sirNuees0NIbG
2	https://youtu.be/783KAB-tuE4 - NPTEL IIT, Madras

Course Outcomes: After completing this course, students should be able to:

CO-1	Understand various activities of process, thread, memory, file and secondary storage components of an Operating System.
CO-2	Apply various scheduling algorithms of process, memory and secondary storage components.
CO-3	Analyze the concepts of inter process communication, deadlocks, memory allocation strategies, page replacement algorithms of OS.
CO-4	Evaluate various algorithms for handling processes, threads, memory allocation strategies and deadlocks.

Course Outcomes	Program Outcomes												PSO's			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	3	3	3	1	3	1	1	1	3	2	2	2	3	2	1	2
CO2	3	3	3	3	3	2	1	1	2	1	1	2	3	3	1	2
CO3	3	3	3	3	3	1	1	1	2	2	2	2	3	3	1	2
CO4	3	3	3	3	3	2	1	1	2	0	0	1	3	3	1	2

Introduce concepts and terminology used in OS

- Explain threading and multithreaded systems
- Illustrate process synchronization and concept of Deadlock
- Introduce Memory and Virtual memory management, File system and storage techniques

Course Title: Design and Analysis of Algorithms	***Course Code: 20CS420***
Credits (L:T:P): 4:0:0	***Contact Hours (L: T: P): 52:0:0***
Type of Course: Lecture	***Category: Professional Core Course***
CIE Marks: 50	***SEE Marks: 100***

Pre-requisite: Data Structures.

Sl.No	Course Objectives
1	To introduce various design techniques and paradigms for designing algorithms.
2	To analyze the performance of various algorithms for their time and space complexity.
3	To identify, choose appropriate design technique and data structures for developing efficient algorithm for a given problem.

Unit No.	Course Content	No. of Hours
1	**Introduction to Algorithms and Problem Solving** Introduction: Notion of Algorithms, Fundamentals of Algorithmic approach to problem Solving, Important Problem Types, Fundamental data Structures. Analysis Framework, Asymptotic Notations and Basic efficiency classes, Mathematical analysis of Recursive and Non recursive algorithms, Examples.	12
2	**Brute force and Exhaustive Search, Divide and Conquer Design Techniques:** Brute Force Approaches: Introduction, Selection Sort, Bubble Sort, Sequential Search, Brute Force String Matching and Exhaustive search. Divide and conquer: General Divide and Conquer, Masters theorem, Recurrence relations, Binary Search, Merge Sort, Quick Sort, Multiplication of large integers and Strassen's Matrices.	10
3	**Decrease and Conquer, Transform and Conquer, Space-Time Tradeoffs** **Decrease-and-Conquer Approaches:** Introduction, Insertion Sort, Depth First Search and Breadth First Search, Topological Sorting. Transform-and-Conquer: Presorting, Balanced Search Trees, Heaps and Heapsort. Space-Time Tradeoffs: Introduction, Sorting by Counting, Input Enhancement in String Matching, Hashing.	10

4	**Dynamic Programming and Greedy Technique** Dynamic programming**:** Computing binomial coefficient, Warshall's and Floyd's Algorithms, Knapsack Problem and Memory Functions. Greedy Technique: Prim's Algorithm, Kruskal's Algorithm Dijkstra's Algorithm, Huffman Trees.	10
5	**Limitations of Algorithmic Power and Coping with Limitations:** Lower-Bound Arguments, Decision Trees, P, NP, and NP-Complete Problems. **Coping With Limitations of Algorithmic Power:** Backtracking: N-Queens Problem, Hamiltonian Circuit Problem, Subset-Sum Problem. **Branch-and-Bound:** Assignment Problem, Knapsack Problem, Traveling Salesman Problem.	10

Text Books:

Sl. No.	Author/s	Title	Publisher Details
1	Anany Levitin	Introduction to The Design & Analysis of Algorithms	3rd Edition, Pearson Education, 2012

Reference Books:

Sl. No.	Author/s	Title	Publisher Details
1	Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein	Introduction to Algorithms	3rd Edition, PHI ,2010
2	Ellis Horowitz, Sartaj Sahni, SanguthevarRajasekaran	Fundamentals of Computer Algorithms	2nd Edition, Universities Press, 2013
3	R.C.T. Lee, S.S. Tseng, R.C. Chang &Y.T.Tsai	Introduction to the Design and Analysis of Algorithms A Strategic Approach	Tata McGraw Hill, 2012
4	Dave and Dave	Design and Analysis of Algorithms	2nd Edition, Pearson

Web Resources:

Sl. No.	Web link
1	https://nptel.ac.in/courses/106101060/
2	https://nptel.ac.in/courses/106106131/

Course Outcomes: After completing this course, students should be able to:

CO-1	Understand various algorithm design techniques and Mathematical models.
CO-2	Apply appropriate data structures and suitable design technique to develop an algorithm for the given problem.
CO-3	Analyse the problem domain, use mathematical analysis model to estimate the running time efficiency of different algorithms for best, average and worst cases.
CO-4	Evaluate a suitable design technique to develop an algorithm for the given problem.

Course Outcomes	Program Outcomes												PSO's			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	P012	PSO1	PSO2	PSO3	PSO4
CO-1	3	3	3	3	3	0	1	1	3	2	3	2	3	3	3	3
CO-2	3	3	3	3	3	0	1	1	2	2	3	2	3	3	3	3
CO-3	3	3	3	3	3	0	1	1	2	2	3	2	3	3	3	3
CO-4	3	3	3	3	3	0	1	1	3	2	3	2	3	3	3	3

Course Title: Data Communication	***Course Code: 20CS430***
Credits (L: T:P): 4:0:0	***Contact Hours (L: T: P): 52:0:0***
Type of Course: Lecture	***Category: Professional Core Course***
CIE Marks: 50	***SEE Marks: 100***

Pre-requisite: Data Structures

SL No	Course Objectives:
1	To introduce the basic concepts of data communication
2	Learn and analyse the working of physical, data link and network layers
3	Learn to solve the problems related to TCP/IP protocols, Line coding, Switching, Error detection and other related protocols.

Unit No.	Course Content	No. of Hours
1	**Introduction:** Data Communications, Networks, Network Types, Standards and Administration, **Networks Models:** Protocol Layering, TCP/IP Protocol suite, The OSI model.	08
2	**Introduction to Physical Layer-1:** Data and Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance. **Digital Transmission:** Digital to Digital Conversion (Only Line coding: Polar, Bipolar and Manchester coding), Analog to Digital conversion (only PCM), Transmission Modes.	10
3	**Bandwidth Utilization:** Multiplexing, **Transmission Media:** Guided Media, Unguided Media, **and Switching:** Introduction, Circuit Switched Networks and Packet switching, Structure of a Switch.	10
4	**Error Detection and Correction:** Introduction, Block Coding, Cyclic Codes: Cyclic Redundancy Checksum**,** Forward Error Correction: Hamming distance, XOR.	12
5	**Data link Layer:Introduction to Data-Link Layer:** Introduction, Link-Layer Addressing, **Data link Services:** DLC services, Data link layer protocols, Point to Point protocol (Framing, Transition phases only**). Media Access control:** Random Access, Controlled Access and Channelization	12

Text Books:

Sl. No.	Author/s	Title	Publisher Details
1	Behrouz A. Forouzan	Data Communications and Networking	5th Edition, Tata McGraw-Hill, 2013

Reference Books:

Sl. No	Author/s	Title	Publisher Details
1	William Stallings	Data and Computer Communication	10th Edition, Pearson Education, 2014
2	Alberto Leon-Garcia and Indra Widjaja	Communication Networks - Fundamental Concepts and Key architectures	2nd Edition Tata McGraw-Hill, Reprint 2017
3	Larry L. Peterson and Bruce S. Davie	Computer Networks – A Systems Approach	5th Edition, Elsevier, 2012
4	Nader F. Mir	Computer and Communication Networks	Pearson Education, 2007

Web Resources:

Sl. No.	Web link
1	https://nptel.ac.in/courses/106105183/
2	https://nptel.ac.in/courses/106/105/106105082/

Course Outcomes: After completing this course, students should be able to:

CO-1	Describe the services offered by physical and data link layers
CO-2	Summarizes network models, signals and conversions, error detection and correction, switching and link protocols
CO-3	Solve problems on TCP/IP protocols, Line coding, Switching requirements, Error detection and Utilization efficiencies of link protocols

Course Outcomes	Program Outcomes												PSO's			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	P012	PSO1	PSO2	PSO3	PSO4
CO-1	3	3	3	1	1	1	1	1	1	2	1	1	2	3	1	1
CO-2	3	3	3	1	1	1	0	1	0	1	1	1	3	3	2	2
CO-3	3	3	3	1	0	1	0	1	1	2	1	1	3	3	1	1

Course title: Software Engineering	***Course Code: 20CS440***
Credits (L:T:P): 4:0:0	***Contact Hours (L: T: P): 52:0:0***
Type of Course: Lecture	***Category: Professional Core course***
CIE Marks: 50	***SEE Marks:100***

Pre-requisite: Nil

Course objectives:

1	To give insight into basics of software engineering methods, practices, models and their appropriate applications.
2	To provide an idea of building requirement model and managing requirement analysis.
3	To furnish knowledge on design concepts and various forms of software architectural styles.
4	To emphasize on software testing approaches, levels and art of debugging.
5	Understanding the project management activities such as planning, estimation and scheduling.

Unit No.	Course Content	No. of Hours
1.	**Software Process** The Nature of Software,The software Process, Software Engineering Practice, A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Agile development: Agility, agile process and principles, Extreme programming.	9
2.	**Software Analysis** Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements, Requirements Analysis.	9
3.	**Software Design** Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model, Software Architecture, Architectural Styles,Architectural design, Component Concepts, Designing Class-Based Components.	12
4.	**Software Testing** Introduction to Quality, Software Quality, A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Validation Testing, System Testing, The Art of Debugging, Software Testing Fundamentals, White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing.	12
5.	**Managing Software Project** The management spectrum: The People, The Product and The Process.The	10

	project planning process, Project resources, Software Project Estimation, Decomposition techniques, Empirical Estimation Models. Project Scheduling: Basic concepts and Principles**,** Defining a task network, Scheduling.	

Text Books:

Sl. No.	Author/s	Title	Publisher Details
1	Roger S Pressman	Software Engineering-A Practitioners approach	8th edition, McGraw-Hill Publication, 2017

Reference Books:

Sl. No.	Author/s	Title	Publisher Details
1	Pankaj Jalote	An Integrated Approach to Software Engineering	3rd edition, 2019Reprint, Narosa Publications
2	Ian Sommerville	Software Engineering	10th edition, Person Education Ltd, 2016
3	Rajib Mall	Fundamentals of Software Engineering	4th edition PHI Publications, 2014
4	Hitesh Mohapatra, Amiya Kumar Rath	Fundamentals of Software Engineering	BPB Publications 2010

Web Resources:

Sl. No.	Web link
1	http://nptel.ac.in/courses/106101061
2	https://nptel.ac.in/courses/106/105/106105182/

Course Outcomes: After completion of the course, students are able to:

CO-1	Explore the concepts of software process models.
CO-2	Analyze and model software requirements.
CO-3	Apprise system design concepts and process.
CO-4	Apprehend and apply software testing strategies.
CO-5	Comprehend software project management activities.

Course Outcomes	Program Outcomes												PSO's			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO-1	3	2	2	2	2	1	1	1	2	2	1	2	3	2	2	2
CO-2	2	3	2	3	2	1	1	1	3	3	1	3	3	2	2	2
CO-3	2	2	3	2	2	1	1	1	2	3	1	3	3	2	2	2
CO-4	2	2	2	2	3	1	1	1	3	3	1	3	3	2	3	2
CO-5	3	2	2	2	3	1	1	1	3	3	3	3	3	2	3	2

Department: Computer Science and Engineering	
Course Title: Theory of Computation	***Course Code: 20CS450***
Credits (L: T:P):3:0:0	***Category: Professional Core Course***
Type of Course: Lecture	***Total Contact Hours: 39:0:0***
CIE Marks: 50	***SEE Marks: 100***

Pre-requisite: Discrete Mathematical Structure.

Sl.No.	Course Objectives
1	Introduce core concepts in Automata and Theory of Computation.
2	Identify different Formal Language Classes and their Relationships.
3	Design finite state machine, Grammars and Recognize for different formal Languages.
4	Prove or disprove theorems in automata theory using their properties.
5	Determine the decidability and intractability of Computational problems.

Unit No.	Course Content	No. of Hours
1	**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; Equivalence and minimization of automata.	08
2	**Regular Expressions and Properties of Regular Languages:** Regular expressions; Converting DFA's to regular expression by eliminating states, Converting Regular Expressions to Finite Automata; Applications of Regular Expressions; Proving languages not to be regular languages; Closure properties of regular Languages.	08
3	**Context-Free Grammars and Languages:** Context-Free Grammars; Parse trees; Applications of Context-Free Grammars; Ambiguity in Grammars and Languages; Normal forms for CFGs.	08
4	**Pushdown Automata** Definition of the Pushdown Automata; The languages of a PDA; Equivalence of PDA's and CFG's; Deterministic Pushdown Automata.	06
5	**Introduction to Turing Machine:**The Turing machine; Programming techniques for Turing Machines, Extension to the basic Turing machine, restricted Turing machine. Undecidability: A language that is not recursively Enumerable, Undecidable problem that is RE, Undecidable Problems about turing Machines, post's correspondence problems, Other Undecidable Problems.	09

Text Books:

Sl. No.	Author/s	Title	Publisher Details
1	John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman	Introduction to Automata Theory, Languages and Computation	3rd Edition, Pearson education, 2007
2	Peter Linz	Finite Automata & Formal Languages	4th edition, Narosa Publication, 2010.
3	John C Martin	Introduction to Languages and Automata Theory	3rd Edition, Tata McGraw- Hill, 2007.
4	Daniel I.A. Cohen	Introduction to Computer Theory	2nd Edition, John Wiley & Sons, 2004
5	Thomas A. Sudkamp	An Introduction to the Theory of Computer Science, Languages and Machines	3rd Edition, Pearson Education, 2006

Web Resources:

Sl. No.	Web link
1	https://nptel.ac.in/courses/106104148/
2	https://nptel.ac.in/courses/106/104/106104028/

1.

Course Outcomes: After completion of the course, students are able to:

CO-1	Understand and explain the concepts of Automata Theory and formal Language.
CO-2	Apply the concepts of theoretical foundations of computing to show the equivalence among different notations of regular and context free languages.
CO-3	Analyze the given language and formulate regular expressions and context free grammars.
CO-4	Identify the class of languages based on Chomsky hierarchy to prove the membership of regular and context free languages.
CO-5	Design and implement automata for regular, context free, and recursively enumerable languages corresponding to real word problem.

Course Outcomes	Program Outcomes												PSO's			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO-1	3	3	0	1	2	0	0	0	1	1	1	1	3	1	2	1
CO-2	3	3	0	2	2	0	0	0	2	1	1	1	3	1	2	1
CO-3	3	3	0	2	2	0	0	0	2	1	1	1	3	1	2	1
CO-4	3	3	0	2	2	0	0	0	2	1	1	1	3	1	2	1
CO-5	3	3	0	2	2	0	0	0	2	1	1	1	3	1	2	1

Course Title: Operating Systems Lab	***Course Code: 20CS46L***
Credits (L:T:P): 0:0:1.5	***Contact Hours (L: T: P): 0:0:39***
Type of Course: Practical	***Category: Professional Core Course***
CIE Marks: 50	***SEE Marks: 50***

Pre-requisite: Computer organization and data structures.

SL.No	Course Objectives:
1	Learn to analyze and develop processschedulingalgorithms.
2	To introduce the concepts of operating system, process creation, synchronization and deadlock avoidance.
3	Formulate and solve various memory management and page replacement schemes.

Week	Programs
1.	i) Exposure to Linux Operating System and Environment ii) Write Shell Scripts for the Following: a) Concatenation of two strings b) Comparison of two strings Maximum of three numbers
2.	Write Shell Scripts for the Following: a) Fibonacci series b) Arithmetic operation using case c) Check whether a given no. is palindrome or not
3.	Write Shell Scripts for the Following: a) Finding largest of N numbers (storing numbers in an array) b) Generating prime numbers. c) Reading two matrices and finding sum
4.	Simulation of ls, rm and grep commands using system calls.
5.	a) Program to illustrate the creation, executing and termination of processes withfork, exec and exit system calls. b) Program to demonstrate the creation of Zombie and Orphan processes. Program to demonstrate the use of I/O system calls for both console and file I/O(read, write, open, lseek)
6.	Simulate the following CPU scheduling algorithms a. FCFS b. SJF c. SJF (Preemptive)
7.	Simulate the following CPU scheduling algorithms a. Priority b. Priority (Preemptive)

	c. Round Robin
8.	i) Write a program to generate and print Fibonacci series with the following requirements: - Parent program should create a child and distribute the task of generating Fibonacci no to its child. - The code for generating Fibonacci no. should reside in different program. - Child should write the generated Fibonacci sequence to a shared memory. - Parent process has to print by retrieving the Fibonacci sequence from the shared memory. a) Implement the above using shmget and shmat b) Implement the above using shm_open and mmap Note: Shared object should be removed at the end in the program. ii. Write a program to generate first N ODD numbers with the following requirements: - Parent program should create a child and distribute the task of generating oddnumbers to its child. - The code for generating odd numbers should reside in different program. - Child should write the odd numbers to a shared memory. - Parent process has to print the odd numbers by retrieving from theshared memory. a) Implement the above using shmget and shmat Note: Shared object should be removed at the end in the program iii. Write a program to generate and print Fibonacci series with the following requirements: - Parent program should create a child and distribute the task of generating Fibonacci no to its child. - The code for generating Fibonacci no. should reside in different program. - Child should write the generated Fibonacci sequence to a shared memory. - Parent process has to print by retrieving the Fibonacci sequence from the shared memory. a) Implement the above using shmget and shmat b) Implement the above using shm_open and mmap Note: Shared object should be removed at the end in the program.
9.	Implement the following file allocation strategies a. First Fit b. Best Fit c. Worst Fit

10.	Implement the following using semaphores a) Producer – Consumer problem b) Reader'swriters' problem
11.	Simulate Bankers algorithm for Deadlock Avoidance.
12.	Simulate the following Page Replacement Algorithms a. FIFO b. LRU **c.** Optimal
13.	**Lab Test/Event**

Reference Books:

Sl. No.	Author/s	Title	Publisher Details
1	Abraham Silberschatz, Peter Baer Galvin, Greg Gagne	Operating System Principles	9th Edition, Wiley India, 2013
2	William Stallings	Operating Systems: Internals and Design Principles	7th Edition,Prentice Hall of India,2017
3	D.M Dhamdhere	Operating systems - A concept-based Approach	4th Edition, Tata McGraw-Hill, 2013
4	P.C.P. Bhatt	Introduction to Operating Systems	Concepts and Practice, 4th Edition, PHI, 2014

Web Resources:

Sl. No.	Web link
1	https://youtu.be/783KAB-tuE4 - NPTEL IIT, Madras
2	https://nptel.ac.in/courses/106108101/

Course Outcomes: After completion of course, the students are able to:

CO-1	Analyze and implement processs chedulingalgorithms.
CO-2	Apply the concepts of operating system and implement process creation, synchronization and deadlock avoidance algorithms.
CO-3	Implement and analyze various memory management and page replacement schemes.

Course Outcomes	Program Outcomes												PSO's			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO-1	3	3	3	3	2	0	0	2	1	1	1	1	3	3	3	3
CO-2	3	3	3	3	2	0	0	2	1	1	1	1	3	3	3	3
CO-3	3	3	3	3	2	0	0	2	1	1	1	1	3	3	3	3

Course Title: Design and Analysis of Algorithms Lab	***Course Code: 20CS47L***
Credits (L: T:P): 0:0:1.5	***Contact Hours (L: T: P): 0:0:39***
Type of Course: Practical	***Category: Professional Core Course***
CIE Marks: 50	***SEE Marks: 50***

Pre-requisite: Data Structures, Design and analysis of algorithms.

Sl No	Course Objectives:
1	Analyze the asymptotic performance of algorithms.
2	Write rigorous correctness proofs for algorithms.
3	Demonstrate a familiarity with major algorithms and data structures.
4	Apply important algorithmic design paradigms and methods of analysis.
5	Synthesize efficient algorithms in common engineering design situations.

Unit No.	List of Programs
1	a) Implement Euclid's, consecutive integer checking and modified Euclid's algorithms to find GCD of two nonnegative integers. b) Sequential search and Binary search algorithms for searching an element in the list.
2	a) Implement Bubble sort algorithm to sort an array of 'N' elements in descending order. Analyze the algorithm to find the order of growth of the algorithm's basic operation count for best case and worst-case inputs and plot the graph for the same. b) Implement Selection sort algorithm to sort an array of N elements in ascending order. Analyze the algorithm to find the order of growth of the algorithm's basic operation count for best case and worst-case inputs and plot the graph for the same.
3	a) Applying suitable design strategy implement **a linear algorithm** to evaluate a given polynomial. Analyze the algorithm to find the order of growth of the algorithm's basic operation count and plot the graph for the same. b) Implement Brute force string matching algorithm to search for a pattern of length 'M' in a text of length 'N' (M<=N). Analyze the algorithm to find the order of growth of the algorithm's basic operation count for best case and worst-case inputs and plot the graph for the same
4	a) Implement Merge sort algorithm to sort an array of 'N' elements in descending order. Analyze the algorithm to find the order of growth of the algorithm's basic operation count for best case and worst-case inputs and plot the graph for the

	same. b) Implement Quick sort algorithm to sort an array of 'N' elements in descending order. Analyze the algorithm to find the order of growth of the algorithm's basic operation count for best case and worst-case inputs and plot the graph for the same
5	a) Implement Recursive binary search algorithm to search for an element in an array of N elements. Analyze the algorithm to find the order of growth of the algorithm's basic operation count for best case and worst-case inputs and plot the graph for the same. Also count and establish the order of growth of function calls. b) Applying divide and conquer technique, implement an algorithm to find largest and smallest element in an array of 'N' elements. Analyze the algorithm to find the order of growth of the algorithm's basic operation count andplot the graph for the same.
6	a) Implement Insertion sort algorithm, analyze its efficiency for worst case and best-case inputs (specify worst case and best-case inputs) and plot the graph b) Implement DFS algorithm to check for connectivity of a graph. If not connected, display the connected components.
7	a) Implement BFS algorithm to check for connectivity of a graph. If not connected, display the connected components. b) Apply DFS algorithm to perform topological sorting.
8	a) Implement source removal algorithm to solve topological sorting problem. b) Implement DFS algorithm to check for acyclicity of a graph. If not, display the cycles.
9	a) Implement heap sort algorithm with bottom-up heap construction. Analyze its efficiency. b) Implement Horspool's String matching.
10	a) Implement Warshall's Algorithm to find the transitive closure of a given directed graph. b) Find All-pair shortest path for a given graph using Floyd's Algorithm.
11	a) Implement 0/1 Knapsack problem using Dynamic Programming. b) Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
12	a) Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm. b) Find shortest paths to other vertices in a weighted connected graph using Dijkstra's algorithm.
13	**Lab Test/Event**

Reference Books:

Sl. No.	Author/s	Title	Publisher Details
1	Anany Levitin	Introduction to The Design & Analysis of Algorithms	3rd Edition, Pearson Education, 2012.
2	Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran	Fundamentals of Computer Algorithms	2nd Edition, Universities Press, 2013.
3	Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein	Introduction to Algorithms	3rd Edition, PHI ,2010
4	R.C.T. Lee, S.S. Tseng, R.C. Chang &Y.T.Tsai	Introduction to the Design and Analysis of Algorithms A Strategic Approach	Tata McGraw Hill, 2012.

Web Resources:

Sl. No.	Web link
1	https://nptel.ac.in/courses/106101060/
2	https://nptel.ac.in/courses/106106131/

Course Outcomes: After completing this course, students should be able to:

CO-1	Analyze the problem domain; Choose the appropriate data structures and design technique based on the problem domain.
CO-2	Implement algorithms and perform analysis with empirical method.
CO-3	Evaluate the performance of different algorithms using different design techniques for solving the same problem.

Course Outcomes	Program Outcomes												PSO's			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO-1	3	3	3	3	3	1	1	1	3	2	3	2	3	3	3	3
CO-2	3	3	3	3	3	1	1	1	3	2	3	2	3	3	3	3
CO-3	3	3	3	3	3	1	1	1	3	2	3	2	3	3	3	3


JSS
SCIENCE AND
TECHNOLOGY
UNIVERSITY
MYSURU