B.Tech. *Computer Engineering* **SEMESTER IV**														
Course No.	**Type**	**Course**	**L**	**T**	**P**	**Credits**	**Evaluation Scheme (Percentage weights)**					**Offering Dept.**	**AICTE COURSE TYPE**
							Theory			**Practical**			
							CA	**MS**	**ES**	**CA**	**ES**		
CECSC09	CC	Operating Systems	3	0	2	4	15	15	40	15	15	CSE	PROGRAM CORE
CECSC10	CC	Theory of Automata & Formal languages	3	1	0	4	25	25	50			CSE	PROGRAM CORE
CECSC11	CC	Software Engineering	3	0	2	4	15	15	40	15	15	CSE	PROGRAM CORE
CEECC12	CC	Data Communication	3	0	2	4	15	15	40	15	15	ECE	ENGG SCIENCES
CEMTC13	CC	Probability and Stochastic Processes	3	1	0	4	25	25	50			MATHS	BASIC SCIENCES
FExxx03*	FE	Elective Foundation	-	-	-	NIL	-	-	-	-	-	-	MANDATORY COURSE
			28 2*			**20**							
2*: The actual weekly load depends upon the elective chosen by the student under FE.													

SYLLABI B. TECH. *Computer Engineering* IV SEMESTER

Course Code	Type	Subject	L	T	P	Credits	TCA	TMS	TES	PCA	PES	Pre-requisites
CECSC09	**CC**	**Operating Systems**	**3**	**0**	**2**	**4**	**15**	**15**	**40**	**15**	**15**	Design and Analysis of Algorithms

COURSE OUTCOMES

1. Understand the function, structure, history of an operating system and the design issues associated with an operating system.
2. Understand the concept of multithreading, process management concepts including scheduling, synchronization and deadlocks.
3. Learn the memory management concepts including virtual memory.
4. Comprehend file system interface and implementation and disk management.
5. Be familiar with protection and security mechanisms.

COURSE CONTENT

Unit 1

Overview: Operating systems – structure, operations, components, types, services, user interfaces. System calls, system programs, system boot.
Process management: Processes – concept, scheduling, operations on processes, interprocess communications. IPC Methods, pipes, popen, pclose functions, Co-Processes, FIFOs, Message Queues, Shared Memory, Stream pipes, Threads – single- and multi-threaded processes.

Unit 2

CPU scheduling – criteria, algorithms, multiple-processor scheduling.
Process synchronization – critical-section problem, semaphores, classic synchronization problems, monitors.

Unit 3

Deadlocks – characterization, deadlock prevention, deadlock avoidance, deadlock detection, prevention, avoidance, recovery from deadlock.
Memory management: Objective and functions, Simple monitor resident program, overlays-swapping, Main memory – memory allocation schemes, paging, segmentation. Virtual memory concept– demand paging, page interrupt fault, page replacement algorithms, segmentation – simple, multilevel, segmentation with paging, frame allocation, thrashing.

Unit 4

Storage management: File system – files and directories, structure and implementation of file systems, mounting and unmounting, storage allocation methods, free-space management. Disk – structure, scheduling, management.

Unit 5

I/o management: i/o hardware, i/o interface, kernel i/o subsystem.
Protection and security: Access matrix, security threats.
Case studies of latest operating systems.

Practical List

Implement these programs in C/C++ using Linux/Unix environment operating system. Maintain hard copy of the same for final assessment.

1. Process creation and termination for operating system (fork, wait, signal, exit etc.).
2. Threads.
3. CPU scheduling algorithms: FCFS, SJF, Round Robin, Preemptive Priority Scheduling.
4. Inter process communication.
5. Critical Section problem.
6. Producer – Consumer problem using bounded and unbounded buffer.
7. Reader Writers problem, Dining Philosophers problem using semaphores.
8. Banker's algorithm.
9. Page replacement algorithms: LRU, LRU-Approximation, FIFO, Optimal.
10. File operation system calls (open, read, close, append etc.)
11. Disk scheduling algorithms: FCFS, SSTF, SCAN, CSCAN, LOOK, CLOOK.

Text Book:

1. Silberschatz, A., Galvin, P. B., and Gagne, G. 2009. "Operating System Principles (8th ed.)", Wiley.

Reference Book:

1. Stallings, W. 2014. "Operating Systems: Internals and Design Principles (8th ed.)", Pearson.
2. Tanenbaum, A. S. 2007. "Modern Operating Systems (3rd ed.)", Pearson.
3. UNIX System Programming Using C++,by Terrence Chan: Prentice Hall India, 1999.
4. Advanced Programming in UNIX Environment, by W. Richard Stevens: 2nd Ed, Pearson Education, 2005.
5. Operating Systems – William Stallings, Pearson Education Asia (2002)
6. Operating Systems - Nutt, Pearson Education Asia (2003)

B.Tech. *Computer Engineering* **SEMESTER IV**

Course No.	Type	Subject	L	T	P	Credits	CA	MS	ES	CA	ES	Pre-requisites
CECSC10	**CC**	**Theory of Automata & Formal**	**3**	**1**	**0**	**4**	**25**	**25**	**50**			**Discrete Structures**

		lan-guages										**Computer Pro-gramming**

COURSE OUTCOMES

1. Students will be able to demonstrate knowledge of basic mathematical models of computation and relate them to the formal languages.
2. Acquire knowledge of Regular Languages, FA, CFG, Push Down Automata and Turing recognizable languages
3. Be able to get a broad overview of the theoretical foundations of computer science
4. Be able to think analytically and intuitively for problem solving situations in related areas of theory of computer science
5. Students will understand the limitations of computers and learn examples of unsolvable problems.

COURSE CONTENT

UNIT I

Finite Automata: Deterministic FA, Non deterministic FA, Regular expressions, Finite Automaton with €- moves, Regular Expression, Regular Languages and Kleene's theorem– Conversion of NFA to DFA, Equivalence of finite Automaton and regular expressions, Arden's Theorem. Myhill Nerode Theorem, Minimization of DFA, Pumping Lemma for Regular sets, Problems based on Pumping Lemma.

UNIT II

Context Free Grammar: Grammar, Types of Grammar, Context Free Grammars and Languages, Derivations, Ambiguity, Relationship between derivation and derivation trees, Simplification of CFG, Elimination of Useless symbols - Unit productions - Null productions, Chomsky normal form (CNF), Greibach Normal form (GNF), Problems related to CNF and GNF.

UNIT III

Pushdown Automata: Moves, Instantaneous descriptions, Deterministic pushdown automata, Equivalence of Pushdown automata and CFL, pumping lemma for CFL, problems based on pumping Lemma.

UNIT IV

Turing Machine: Definitions of Turing machines, Computable languages and functions, Techniques for Turing machine construction, Multi head and Multi tape Turing Machines, The Halting problem, Partial Solvability, Problems about Turing machine- Chomsky hierarchy of languages.

UNIT V

Difficult problems: Unsolvable Problems and Computable Functions, Primitive recursive functions, Recursive and recursively enumerable languages, Universal Turing machine, Measuring and classifying complexity - Tractable and Intractable problems,

SUGGESTED READINGS

1. Hopcroft J.E., Motwani R. and Ullman J.D, "Introduction to Automata Theory, Languages and Computations", Second Edition, Pearson Education.
2. John C Martin, "Introduction to Languages and the Theory of Computation", Third Edition, Tata McGraw Hill Publishing Company, New Delhi
3. Marvin L. Minsky "Computation: Finite and Infinite" – Prentice Hall, 1967
4. Michael Sipser "Introduction to the Theory of Computation" , Third Edition, 2012 Cengage Learning
5. Peter Lenz – An Introduction to Formal languages and Automata – 3rd Edition Narosa, 2003
6. Thomas A. Sukamp – An introduction to the theory of computer science languages and machines – 3rd edition, Pearson Education, 2007.
7. G E Reevsz "Introduction to Formal Languages" TMH, 2000

B.Tech. *Computer Engineering* **SEMESTER IV**

Course No.	Type	Subject	L	T	P	Credits	CA	MS	ES	CA	ES	Pre-requisites
CECSC11	**CC**	**Software Engineering**	**3**	**0**	**2**	**4**	**15**	**15**	**40**	**15**	**15**	Design and Analysis of Algorithms

COURSE OUTCOMES

1. To appreciate the fact that software development cannot be done in an adhoc fashion and has to follow a disciplined systematic approach for timely development of software within budget using suitable Process model and techniques
2. To learn various techniques for Requirements Elicitation and Specification in order to develop SRS for a problem domain

3. To learn Different techniques for software project management like Feasibility Analysis, Cost and Effort Estimation, Scheduling a project etc and Architecture Styles
4. To learn to test a software using suitable verification and validation testing techniques
5. To learn about different Software Quality frameworks, Software metrics, Software Reliability, Risk management, software maintenance etc.

COURSE CONTENTS

Unit 1

Introduction: Introduction to software engineering, Importance of software, The Software evolution, Software characteristics, Software components, Software applications, Crisis-Problem and causes. Difference between software engineering and system engineering

Software Process Models: Waterfall model, Evolutionary Models, prototyping, V Model, Spiral model Incremental Model, RAD Model etc.

Introduction to Agile models like Scrum, Extreme Programming, Feature Driven Development, Crystal etc., Comparison between Traditional and Agile models

Requirement Engineering: Different Types of Requirements: Functional, Non Functional and Domain Requirements in detail, Requirement elicitation Techniques like interviews, questionnaire, brainstorming, JAD, Scenario, Mind mapping, Requirement workshop, Prototyping, CRC Cards etc. Requirements Management, Writing SRS as per IEEE standard, Quality characteristics of SRS

Unit 2

Requirements Specification: Difference between structured and Object Oriented Analysis, Different views of modeling, ER diagram, Data flow diagrams, State Transition Diagrams, data Dictionary, techniques for describing process specifications, Introduction to Unified Modelling Language(UML), Introduction to Use Case Diagrams in detail

Unit 3

System Design: Types of Coupling and Cohesion, Deriving structure chart from DFD in case of structured analysis.

Software Architecture: Importance of Software architecture, Different views of Software Architecture, Popular Architecture Styles

Software project Management: Project Management Process, System Request, Feasibility Analysis in detail Cost estimation Models like COCOMO, Function Point Analysis, Putnam estimation model etc., Project scheduling, Finding Critical Path, staffing,

Introduction to software configuration management

Unit 4

Software Testing: Difference between verification and validation testing, Introduction to verification and Validation testing techniques, Levels of testing, , Alpha and Beta testing, System testing.

Unit 5

Software Quality : Software Quality Models like McCall's Quality model, Quality frameworks like Capability Maturity Model, ISO9001, Software quality metrics, Software Reliability

Software Maintenance : Different Types of maintenance

Risk management.

Current trends and future directions for Software Engineering

Practical:

1. Choose a problem domain and textually write the Functional and Non Functional requirements of the domain. While writing make use of elicitation techniques discussed in the class
2. For a Problem domain develop a Mind-Map
3. Draw the static View of the Problem domain of exercise 1 using ER diagram
4. Draw the functional view of the problem domain of exercise 1 using Data Flow Diagram
5. Write down the data dictionary and process specifications of the processes of DFD using different techniques discussed in the class
6. Draw the Dynamic view of working of an ATM Machine or Microwave Oven
7. Generate SRS as per IEEE std-830 using structured analysis techniques for the problem domain you have described in exercise 1
8. Develop the Structure chart from the DFD developed in exercise 4.
9. Develop a Use Case Model(use Case diagram and Use case Narratives) for a problem domain
10. Perform Effort estimation activity and Implement Critical path method using an open source Tool for a case study.
11. Study of tools for Configuration management and Cost estimation models

SUGGESTED READINGS

1. R . S. Pressman, Bruce R. Maxim "Software Engineering – A practitioner's approach", McGraw Hill Int. Edition, Eight Edition, 2019
2. Sommerville, "Software Engineering", 10th Edition, Pearson, 2017
3. Sangeeta Sabharwal, "Software Engineering: Principles and Techniques", Second Edition, Published by New Age International Publishers, 2020
4. Rajib Mall, "Fundamentals of Software Engineering" , PHI learning Pvt Ltd, 2018

Course Code	Type	Subject	L	T	P	Credits	CA	MS	ES	CA	ES	Pre-requisites
CEECC12	**CC**	**Data Communication**	**3**	**0**	**2**	**4**	**15**	**15**	**40**	**15**	**15**	**None**

COURSE OUTCOMES

1. To introduce students about different digital modulation schemes.
2. To introduce the students the functions of different layers of networking.
3. To introduce various types of access control methods.
4. To make students to get familiarized with different protocols and network components.
5. To introduce the students about basic queuing models

COURSE CONTENT

UNIT-I

Digital Communication: Sampling theorem (Instantaneous Sampling, Natural Sampling and Fla Top Sampling), PAM, PPM, PWM, Quantization noise, PCM, Binary Modulation: ASK, PSK, FSK MSK, DPSK, QPSK and their probability of error calculation.

UNIT-II

Data Communications: Review of Error Detection and Correction codes. Need of line coding, Line coding scheme: Unipolar, Polar, Bipolar and Multilevel Encoding, Network and Protocol Architecture, Reference Model ISO-OSI, TCP/IP-Overview, topology, DTE-DCE interface, interface standards, modems, cable modem, transmission media. Switching: Circuit switching (space-division, time division and space-time division), packet switching (virtual circuit and Datagram approach), message switching,

UNIT-III

Data Link Layer: Data Link Control and Protocols: Flow and Error Control, Stop-and-wait ARQ. Sliding window protocol, Go-Back-N ARQ, Selective Repeat ARQ, HDLC, Point-to –Point Access: PPP Point –to- Point Protocol, PPP Stack, IEEE standard 802.3 & 802.11 for LANS, high speed LANs, Token ring, Token Bus, FDDI based LAN, Network Devices-repeaters, hubs, switches bridges.

UNIT-IV

Medium Access Sub layer: Channel allocation problem, multiple access protocols (ALOHA, CSMA and CSMA/CD)

Network Layer: Design issues, Routing algorithms, Congestion control algorithms, Host to Host Delivery: Internetworking

UNIT-V

Queuing Theory: Finite Markov Chain –Discrete and continuous time Markov chains, Classification of states, Limiting distribution, Birth and death process, Poisson process, Steady state and transient distributions, Simple Markovian queuing models (M/M/1, M/M/1/N).

List of Experiments

1. Introduction to MATLAB
   a. Matrix computation.
   b. To Plot Sine Wave of frequency 200 Hz.
   c. To plot a pulse of width 10.
   d. Plot the spectrum (Amplitude and phase) 0f the pulse generated in 3.
2. Uniform random number and plot its density function. Find its mean and variance.
3. Generate Gaussian distributed random number and plot its density function. Find its mean and variance.
4. Compute the Signal to quantization Noise ratio of Uniform Quantization. Plot SNQR versus Quantization levels.
5. Compute the Signal to quantization Noise ratio of Non-Uniform Quantization. Plot SNQR versus Quantization levels.
6. Study of passband digital communication technique BPSK. Calculate the BER of BPSK modulated signal.
7. Given is a linear block code with the generator matrix G

1 1 0 0 1 0 1

a. G = 0 1 1 1 1 0 0 1 1 1 0 0 1 1

   a. Calculate the number of valid code words N and the code rate RC. Specify the complete Code set C.
   b. Determine the generator matrix G′ of the appropriate systematic (separable) code C'.
   c. Determine the syndrome table for single error.
8. To generate a M/M/1 Queue having infinite buffer space with parameters (λ, μ) and

plot the average delay per packet vs λ/μ.

9. To generate a M/M/1 Queue having finite buffer space with parameters (λ, μ) and plot blocking probability with respect to variation with buffer space.
10. To simulate STOP and WAIT protocol using M/M/1 queuing system and plot average delay per packet vs λ/μ.
11. To simulate SLIDING WINDOW protocol and evaluate its performance with variation of window size.
12. Observe and measure the performance of TOKEN BUS MAC Layer protocols by changing the network load, distance between the nodes.
13. Observe and measure the performance of ALOHA protocol by changing the network load, distance between the nodes.
14. Observe and measure the performance of CSMA protocols by changing the network load, distance between the nodes.
15. Observe and measure the performance of CSMA/CD protocols by changing the network load, distance between the nodes.

Text Book:

1. A. S. Tannenbum, D. Wetherall, "Computer Networks", Prentice Hall, Pearson, 5 th Ed [T2] Behrouz A. Forouzan, "Data Communications and Networking", Tata McGraw-Hill, 4th Ed

Reference:

1. Fred Halsall, "Computer Networks", Addison – Wesley Pub. Co. 1996.
2. Larry L, Peterson and Bruce S. Davie, "Computer Networks: A system Approach", Elsevier, 4 th Ed
3. Tomasi, "Introduction To Data Communications & Networking", Pearson 7th impression 2011
4. William Stallings, "Data and Computer Communications", Prentice Hall, Imprint of Pearson, 9 th Ed.
5. Zheng , "Network for Computer Scientists & Engineers", Oxford University Press
6. Data Communications and Networking: White, Cengage Learning

Course No.	Title of the Course	Course Structure	Pre-Requisite
CEMTC13	Probability and Stochastic Processes	3L-1T-0P	None

COURSE OUTCOMES (CO)

1. To understand the detailed concept of probability and applications.
2. To know about Continuous Frequency distribution.
3. To know about MGF and Method of Least square.
4. To understand the concept of large samples.
5. To understand sampling theory for small samples and inference.

COURSE CONTENT:

UNIT-1

Probability: Mathematical and Statistical definitions and problems, Marginal probability, Random variables, discrete and continuous random variables, Mathematical Expectation, Moments, Central moments, Kurtosis.

UNIT-2

Important Theoretical Distributions: Review of continuous and discrete probability distributions, Negative binomial distribution, Fitting of standard distributions, Fitting of Normal distribution by method of areas and method of ordinates, Hypergeometric distributions, Multinomial distribution, Rectangular distribution, Beta distribution of first and second kind, Gamma distribution, Cauchy's distribution, Geometrical probability, Tchebycheff's and Markov's inequalities.

UNIT-3

MGF and Method of Least Square: Change of origin and scale in MGF, moment generating functions of standard distributions (Poisson, Binomial, Exponential, Uniform, Normal, Gamma, chi square), Cumulants, characteristic function, Weak law of large numbers, Central limit theorem. **Method of least squares**: Fitting of straight lines, parabola and exponential curves.

UNIT-4

Simple sampling of attributes: Large samples, mean and S. D. in simple sampling of attributes, Test of significance for large samples, Standard error, Type I and II errors, Null hypothesis, Confidence limits, Chi-square distribution, Degree of freedom, Level of significance, Test of goodness of fit, Test of independence, Coefficient of contingency, Yate's correction for continuity.

UNIT-5

Sampling of variables and Inference: Small samples, t-distribution, test of significance of the mean of random sample from normal population, F-distribution, Relationship between t, F and chi square distributions, **Inference:** Point estimation, interval estimation, properties of good estimator, Maximum likelihood parameter.

Recommended Books:

1. An Introduction to Probability Theory and Its Applications, Vol. 1 (Wiley Series in Probability and Statistics) by W. Feller, 1968
2. Advanced Engineering Mathematics by Erwin Kreyszig (Wiley Publication), 2020
3. Probability & Statistics- SOS by Spiegel, McGraw Hill, 2010
4. Probability and Statistics for Engineers by Anthony J. Hayter (Cengage Learning), 2013
5. Mathematical Statistics, Krishna Prakashan Media by J. K. Goyal and J. N. Sharma, 2014