E.G.S. PILLAY ENGINEERING COLLEGE

(Autonomous)

Approved by AICTE, New Delhi | Affiliated to Anna University, Chennai

Accredited by NAAC with 'A' Grade | Accredited by NBA (CSE, EEE, MECH)

NAGAPATTINAM – 611 002

M.E. COMPUTER SCIENCE AND ENGINEERING

Full Time Curriculum and Syllabus

First Year – First Semester

Course Code	Course Name	L	T	P	C	Maximum Marks		
						CA	ES	Total
Theory Course								
1701CP101	Applied Probability and Statistics	2	2	0	3	40	60	100
1702CP102	Advanced Data Structures and Algorithms	3	0	0	3	40	60	100
1702CP103	Advanced Computer Architecture	3	0	0	3	40	60	100
1702CP104	Advanced Operating System	3	0	0	3	40	60	100
1702CP105	Design and Management of Computer Networks	3	0	0	3	40	60	100
	Elective I	3	0	0	3	40	60	100
Laboratory Course								
1704CP106	Advanced Data Structures Laboratory	0	0	4	2	50	50	100
1704CP107	Case Study – Operating Systems Design	0	0	2	1	50	50	100
1704CP108	Communication Skill Lab I	0	0	2	1	100	0	100

L – Lecture | T – Tutorial | P – Practical | C – Credit | CA – Continuous Assessment | ES – End Semester

1701CP101	APPLIED PROBABILITY AND STATISTICS	L	T	P	C
		3	2	0	4

COURSE OBJECTIVES:

1. To introduce the basic concept of Probability function
2. To enable the students in handling Estimation and Testing of Hypothesis
3. To learn the Application of Statistics in Engineering Decision Making

UNIT I INTRODUCTION **9 Hours**

Basic definitions and rules for Probability- Properties-Conditional Probability- Independent Events-Mutually exclusive Events- Total Probability- Baye'Theorem

UNIT II RANDOM VARIABLES **9 Hours**

One dimensional Random Variable- Moments- Moment Generating Function- Functions of Random Variable- Two Dimensional Random Variable - Correlation

UNIT III ESTIMATION THEORY **9 Hours**

Estimation: Point and Interval estimates for population parameters of large sample and small samples, determining the sample size- unbiased Estimators- Maximum Likelihood Estimation-Curve Fitting by Principle of Least square

UNIT IV TESTING OF HYPOTHESIS- PARAMETRIC TESTS **9 Hours**

Hypothesis testing: one sample and two sample tests for means and proportions of large samples z-test, one sample and two sample tests for means of small sample t-test, F-test for two sample standard deviations. ANOVA one and two way.

UNIT V NON PARAMETRIC TESTS **9 Hours**

Chi-square test for single sample standard deviation. Chi-square tests for independence of attributes and goodness of fit. Sign test for paired data. Rank sum test. Comparing two populations. Mann – Whitney U test and Kruskal Wallis test.

TOTAL: 45 + 15 HOURS

FURTHER READING:

Queuing Theory
Probability and statistics

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1 Acquire knowledge in basic concepts of Probability
CO2 Deal with one dimensional and two dimensional Random Variable
CO3 Estimate the sample size and prediction of unknown values
CO4 Solve Parametric and non parametric statistical problem
CO5 Apply statistical techniques for solving Engineering problems

REFERENCES:

1. Jay L. Devore, "Probability and Statistics For Engineering and the Sciences",Thomson and Duxbury, 2002.
2. Richard Johnson. "Miller & Freund's Probability and Statistics for Engineer", Prentice – Hall,Seventh Edition, 2007.
3. Richard A. Johnson and Dean W. Wichern, "Applied Multivariate Statistical Analysis", Pearson Education, Asia, Fifth Edition, 2002.
4. Gupta S.C. and Kapoor V.K."Fundamentals of Mathematical Statistics", Sultan and Sons, 2001.
5. Dallas E Johnson , "Applied Multivariate Methods for Data Analysis", Thomson an Duxbury press,1998.

1702CP102	ADVANCED DATA STRUCTURES AND ALGORITHMS	L	T	P	C
		4	0	0	4

COURSE OBJECTIVES:

1. To understand the implementation and use of advanced data structures.
2. To learn how to analyze the space and time requirements of a given algorithm
3. To design efficient algorithms using algorithmic techniques.

UNIT I COMPLEXITY ANALYSIS AND ELEMENTARY DATA STRUCTURES 9 Hours

Asymptotic notations – Properties of big oh notation – Asymptotic notation with several parameters – Conditional asymptotic notation – Amortized analysis – NP Completeness - Arrays – Linked lists – Trees.

UNIT II HEAP STRUCTURES AND AMORTIZED ANALYSIS 9 Hours

Min-max heaps – D-Heaps – Leftist heaps –Binomial heaps – Fibonacci heaps – Skew heaps - Lazy binomial heaps- Amortized analysis – Binomial heaps – Skew heaps – Fibonacci heaps

UNIT III SEARCH STRUCTURES 9 Hours

Binary search trees – AVL trees – 2-3 trees – 2-3-4 trees – Red-black trees – B-trees – Splay trees- Hashing and collision resolution.

UNIT IV GREEDY AND DIVIDE AND CONQUER 9 Hours

Knapsack problem- Minimum spanning trees: Prim's algorithm - Kruskal's algorithm -Tree-vertex splitting – Job sequencing with deadlines – Optimal storage on tapes - Quicksort – Strassen's matrix multiplication – Convex hull.

UNIT V DYNAMIC PROGRAMMING AND BACKTRACKING 9 Hours

Multistage graphs – 0/1 knapsacks using dynamic programming – Flow shop scheduling – 8-queens problem – Graph coloring – Knapsack using backtracking- Hamiltonian cycles

TOTAL: 45 HOURS

FURTHER READING:

Design and Analysis of Algorithm
Advanced Backtracking Techniques

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1 Understand the properties of various data structures. Analyze different algorithm design techniques.

CO2 Design and employ appropriate data structures for solving real time applications.

REFERENCES:

1. Mark Allen Weiss, *Data Structures and Algorithms in C++*, Pearson, 2009
2. E. Horowitz, S. Sahni and S. Rajasekaran, *Computer Algorithms / C++*, University Press, 2007.
3. Adam Drozdex, *Data Structures and* algorithms in C++. New Delhi: Thomson learning, 2006.
4. Gupta S.C. and Kapoor V.K."Fundamentals of Mathematical Statistics", Sultan and Sons, 2001.
5. T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein, *Introduction to Algorithms*, Prentice hall of India, 2003.

1702CP103	ADVANCED COMPUTER ARCHITECTURE	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES:

1. To introduce the fundamental techniques based on parallel processing
2. To develop the foundations for analyzing the benefits of design options in computer architecture
3. To give experience of the application of the various computing techniques.

UNIT I PIPE LINING AND ILP **9 Hours**

Fundamentals of computer design - Measuring and reporting performance - Instruction level parallelism and its exploitation - Concepts and challenges - Overcoming data hazards with dynamic scheduling –Dynamic branch prediction – Speculation-Multiple issue processors.

UNIT II ADVANCED TECHNIQUES FOR EXPLOITING ILP **9 Hours**

Compiler techniques for exposing ILP - Limitations on ILP for realizable processors - Hardware versus software Speculation - Multithreading: Using ILP support to exploit Thread-level parallelism -Performance of advanced multiple issue processors-Efficiency in advanced multiple issue processors.

UNIT III MULTI PROCESSORS **9 Hours**

Symmetric and distributed shared memory architectures – Cache coherence issues - Performance Issues – Synchronization issues – Models of memory consistency - Interconnection networks – Buses, crossbar-Multi-stage switches.

UNIT IV MEMORY HIERARCHY **9 Hours**

Introduction - Optimizations of cache performance - Memory technology and optimizations - Protection: Virtual memory and virtual machines-Design of memory hierarchies.

UNIT V STORAGE SYSTEMS **9 Hours**

Advanced topics in disk storage- Definition and examples of real faults and failures-I/O performance, reliability measures and benchmarks-A Little queuing theory.

TOTAL:45 HOURS

FURTHER READING:

Advanced ILP and TLP Technologies
Advanced Multiprocessors

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1 Analyze the working principle of different ILP and TLP techniques
CO2 Demonstrate the concepts of multiprocessor architecture
CO3 Identify the need of cache and virtual memory.

REFERENCES:

1. John L. Hennessey and David A. Patterson, *Computer Architecture–A quantitative approach*. Noida:Morgan Kaufmann / Elsevier, 2012.
2. William Stallings, *Computer Organization and Architecture–Designing for Performance*. New Delhi: Pearson Education, 2006.
3. David E. Culler and Jaswinder Pal Singh, *Parallel Computing Architecture: A hardware/ softwareapproach*. Noida: Morgan Kaufmann / Elsevier, 1999.

1702CP104	ADVANCED OPERATING SYSTEMS	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES:

1. To learn the fundamentals of Operating Systems.
2. To gain knowledge on Distributed operating system concepts that includes architecture, Mutual exclusion algorithms, Deadlock detection algorithms and agreement protocols
3. To gain insight on to the distributed resource management components viz. the algorithms for implementation of distributed shared memory, recovery and commit protocols
4. To know the components and management aspects of Real time, Mobile operating systems

UNIT I FUNDAMENTALS OF OPERATING SYSTEMS 9 Hours

Overview – Synchronization Mechanisms – Processes and Threads - Process Scheduling – Deadlocks: Detection, Prevention and Recovery – Models of Resources – Memory Management Techniques.

UNIT II DISTRIBUTED OPERATING SYSTEMS 9 Hours

Issues in Distributed Operating System – Architecture – Communication Primitives – Lamport's Logical clocks – Causal Ordering of Messages – Distributed Mutual Exclusion Algorithms – Centralized and Distributed Deadlock Detection Algorithms – Agreement Protocols.

UNIT III DISTRIBUTED RESOURCE MANAGEMENT 9 Hours

Distributed File Systems – Design Issues - Distributed Shared Memory – Algorithms for Implementing Distributed Shared memory–Issues in Load Distributing – Scheduling Algorithms – Synchronous and Asynchronous Check Pointing and Recovery – Fault Tolerance – Two-Phase Commit Protocol – Non blocking Commit Protocol – Security and Protection.

UNIT IV REAL TIME AND MOBILE OPERATING SYSTEMS 9 Hours

Basic Model of Real Time Systems - Characteristics- Applications of Real Time Systems – Real Time Task Scheduling - Handling Resource Sharing - Mobile Operating Systems –Micro Kernel Design - Client Server Resource Access – Processes and Threads - Memory Management – File system.

UNIT V CASE STUDIES 9 Hours

Linux System: Design Principles - Kernel Modules - Process Management Scheduling – Memory Management - Input-Output Management - File System - Interprocess Communication. iOS and Android: Architecture and SDK Framework - Media Layer - Services Layer - Core OS Layer – File System.

TOTAL: 45 HOURS

FURTHER READING:

1. Advanced Mobile Operating Systems
2. Real Time Operating Systems

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

- CO1 Discuss the various synchronization, scheduling and memory management issues
- CO2 Demonstrate the Mutual exclusion, Deadlock detection and agreement protocols of Distributed operating system
- CO3 Discuss the various resource management techniques for distributed systems
- CO4 Identify the different features of real time and mobile operating systems
- CO5 Install and use available open source kernel
- CO6 Modify existing open source kernels in terms of functionality or features used

REFERENCES:

1. Mukesh Singhal and Niranjan G. Shivaratri, "Advanced Concepts in Operating Systems – Distributed, Database, and Multiprocessor Operating Systems", Tata McGraw-Hill, 2001.
2. Abraham Silberschatz; Peter Baer Galvin; Greg Gagne, "Operating System Concepts", Seventh Edition, John Wiley & Sons, 2004.
3. Daniel P Bovet and Marco Cesati, "Understanding the Linux kernel", 3rd edition, O'Reilly, 2005.
4. Rajib Mall, "Real-Time Systems: Theory and Practice", Pearson Education India, 2006.
5. Neil Smyth, "iPhone iOS 4 Development Essentials – Xcode", Fourth Edition, Payload media, 2011.

1702CP105	DESIGN AND MANAGEMENT OF COMPUTER NETWORKS	L	T	P	C
		3	0	2	4

COURSE OBJECTIVES:

1. To learn definitions of network analysis, architecture, and design
2. To study about different types of requirements from the user, application, device and network components.
3. To learn about how to group requirements together and to map the locations of applications and devices.
4. To learn how to identify and characterize traffic flows
5. To develop internal and external relationships within and between major functions like addressing and routing

UNIT I INTRODUCTION TO NETWORK MANAGEMENT 9 Hours

Overview of Analysis, Architecture and Design Process-System Methodology, Service methodology, Service Description - Service characteristics - Performance Characteristics - Network supportability - Requirement analysis – User Requirements – Application Requirements – Device Requirements – Network Requirements – Other Requirements - Requirement specification and map.

UNIT II REQUIREMENTS ANALYSIS 9 Hours

Requirement Analysis Process – Gathering and Listing Requirements- Developing service metrics – Characterizing behavior – Developing RMA requirements – Developing delay Requirements - Developing capacity Requirements - Developing supplemental performance Requirements – Requirements mapping – Developing the requirements specification

UNIT III FLOW ANALYSIS 9 Hours

Individual and Composite Flows – Critical Flows - Identifying and developing flows – Data sources and sinks – Flow models- Flow prioritization – Flow specification algorithms – Example Applications of Flow Analysis

UNIT IV NETWORK ARCHITECTURE 9 Hours

Architecture and design – Component Architectures – Reference Architecture – Architecture Models – System and Network Architecture – Addressing and Routing Architecture – Addressing and Routing Fundamentals – Addressing Mechanisms – Addressing Strategies – Routing Strategies – Network Management Architecture – Network Management Mechanisms Performance Architecture – Performance Mechanisms – Security and Privacy Architecture – Planning security and privacy Mechanisms

UNIT V NETWORK DESIGN 9 Hours

Design Concepts – Design Process - Network Layout – Design Traceability – Design Metrics –Logical Network Design – Topology Design – Bridging, Switching and Routing Protocols- Physical Network Design – Selecting Technologies and Devices for Campus and Enterprise Networks – Optimizing Network Design

TOTAL: 45 HOURS

FURTHER READING:

Advanced Network Requirements
Wireless Network Technologies

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1 Gather, derive, define and validate real requirements for the specified network.
CO2 Implement how and where addressing and routing, security, network management, and performance are required in the network.
CO3 Evaluate and select vendors, vendor products, and service providers for the project.
CO4 Develop traceability between requirements, architecture decisions, and design decisions.
CO5 Apply routing protocols (RIP/RIPv2, OSPF, BGP-4, MPLS), as well as classful and classless IP addressing mechanisms.

REFERENCES:

1. Network Analysis, Architecture, and Design By James D. McCabe, Morgan Kaufmann, Third Edition, 2007.ISBN-13: 978-0123704801
2. Computer Networks: A Systems Approach by Larry L. Peterson, Bruce S. Davie - 2007, Elsevier Inc.
3. Top-down Network Design: [a Systems Analysis Approach to Enterprise Network Design] By Priscilla Oppenheimer, Cisco Press , 3rd Edition, ISBN-13: 978-1-58720- 283-4 ISBN-10: 1- 58720-283-2
4. Integrated Management of Networked Systems: Concepts, Architectures, and Their Operational

Application (The Morgan Kaufmann Series in Networking), Heinz-GerdHegering, Sebastian Abeck, and Bernhard Neumair, 1999.
5. . "Network Design and Management" – by Steven T.Karris, Orchard publications, Second edition, Copyright 2009, ISBN 978-1-934404-15-7
6. Network Design, Management and Technical Perspective", Teresa C. Mann-Rubinson and KornelTerplan, CRC Press, 1999
7. . "Ethernet Networks-Design, Implementation, Operation and Management by Gilbert Held, John Wiley and sons, Fourth Edition
8. . James Kurose and Keith Ross, "Computer Networking: A Top-Down Approach Featuring the Internet", 1999

1704CP106 ADVANCED DATA STRUCTURES LABORATORY

L	T	P	C
0	0	4	2

COURSE OBJECTIVES:

1. To learn to implement iterative and recursive algorithms
2. To learn to design and implement algorithms using hill climbing and dynamic programming techniques
3. To learn to implement shared and concurrent objects

LIST OF EXPERIMENTS:

1. Implementation of Binary Search Tree
2. Implementation of Fibonacci Heaps
3. Implementation of Red-Black tree
4. Implementation of Spanning Tree
5. Implementation of Shortest Path Algorithms
6. Implementation of Graph Traversals
7. Implementation of Greedy Algorithms
8. Implementation of Approximation Algorithms

TOTAL: 45 HOURS

ADDITIONAL EXPERIMENTS:

Tree Traversals
Inverse Graph

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1 Design and apply iterative and recursive algorithms
CO2 Design and implement algorithms using the hill climbing and dynamic programming and recursive backtracking techniques
CO3 Design and implement optimization algorithms for specific applications
CO4 Design and implement randomized algorithms
CO5 Design appropriate shared objects and concurrent objects for applications

REFERENCES:

1. Jeff Edmonds, "How to Think about Algorithms", Cambridge University Press, 2008.
2. M. Herlihy and N. Shavit, "The Art of Multiprocessor Programming", Morgan Kaufmann, 2008
3. Steven S. Skiena, "The Algorithm Design Manual", Springer, 2008
4. Peter Brass, "Advanced Data Structures", Cambridge University Press, 2008
5. S. Dasgupta, C. H. Papadimitriou, and U. V. Vazirani, "Algorithms" , McGrawHill, 2008
6. J. Kleinberg and E. Tardos, "Algorithm Design", Pearson Education, 2006
7. T. H. Cormen, C.2 E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms", PHI Learning Private Limited, 2012

1704CP107	CASE STUDY – OPERATING SYSTEMS DESIGN	L	T	P	C
		0	0	2	1

COURSE OBJECTIVES:

1. To develop capabilities to work at systems level
2. To learn about issues in designing and implementing modern operating systems
3. To understand team formation, team issues, and allocating roles and responsibilities
4. To make effective presentations on the work done
5. To develop effective written communication skills

LIST OF EXPERIMENTS:

1. Development of a reasonably sized dynamically loadable kernel module for Linux kernel
2. Study educational operating systems such as Minix (http://www.minix3.org/), Weenix (http://weenix.cs.brown.edu/mediawiki/index.php/Weenix) and develop reasonably sized interesting modules for them
3. Study the Android open source operating system for mobile devices (http://source.android.com/) and develop / modify some modules.
4. Study any embedded and real-time operating system such as eCos (http://ecos.sourceware.org/) and develop / modify some modules.

TOTAL: 45 HOURS

ADDITIONAL EXPERIMENTS:

Embedded Operating Systems
Real Time Operating Systems

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1 Develop assigned modules of operating systems design carrying out coding, testing, and documentation work involved.
CO2 Describe team issuses and apply suitable methods to resolve the same.
CO3 Demonstrate individual competence in building medium size operating system components.
CO4 Demonstrate ethical and professional attributes of a computer engineer
CO5 Prepare suitable plan with clear statements of deliverables, and track the same.
CO6 Make individual presentation of the work carried out.
CO7 Prepare well-organized written documents to communicate individual work accomplished.

REFERENCES:

1. Prepare well-organized written documents to communicate individual work accomplished.
2. Mukesh Singhal and Niranjan G. Shivaratri, “Advanced Concepts in Operating Systems – Distributed, Database, and Multiprocessor Operating Systems”, Tata McGraw-Hill, 2001.
3. T. W. Doeppner, “Operating Systems in Depth: Design and Programming”, Wiley, 2010.
4. S. Tanenbaum and A. S. Woodhull, “Operating Systems Design and Implementation”, Third
5. Edition, Prentice Hall, 2006.
6. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, “Operating System Concepts”, Ninth Edition, John Wiley & Sons, 2012.
7. Daniel P. Bovet and Marco Cesati, “Understanding the Linux kernel”, 3rd edition, O’Reilly, 2005.
8. Rajib Mall, “Real-Time Systems: Theory and Practice”, Pearson Education India, 2006.

1704CP108	COMMUNICATION SKILLS LAB I (Common to all M.E Progarmmes)	L	T	P	C
		0	0	2	1

COURSE OBJECTIVES:

1. To acquire skills for using English in workplace effectively.
2. To communicate for essential business needs.
3. To prepare students for taking BEC Vantage level examination which is an International Benchmark for English language proficiency of Cambridge English Language Assessment

LIST OF EXPERIMENTS:

1. GRAMMAR AND VOCABULARY

Forming asking complex questions – expressing purpose and function –modal verbs – impersonal passive voice– Reported speech – cause and effect – relative pronouns – expressions followed by – *ing* forms– acronyms – marketing terms / vocabulary – financial terms – collocations – discourse markers

2. LISTENING

Purposes of listening – features of listening texts – potential barriers to listening – specific listening skills – strategies to use when listening– distinguishing relevant from irrelevant information – gap filling exercise – multiple-choice options – note completion – matching and multiple choice questions – listening for specific information, gist, topic, context and function.

3. SPEAKING

Word and sentence stress – clear individual sounds – turn taking – initiating and responding - intonation patterns – pronunciation – mother tongue intrusion– conversation practice – turn-taking and sustaining the interaction by initiating and responding appropriately- Public Speech – Lectures.

4. READING

Purposes of reading – potential barriers to reading – paraphrasing – identifying facts and ideas – skimming and scanning for information – matching statements with texts– spotting reference words – understanding text structure – understanding the ideas in a text – distinguishing between the correct answer and the distracter – understanding cohesion in a text – deciphering contextual meaning of words and phrases – cloze – proof reading - transcoding.

5. WRITING

Paragraphing a text – using appropriate connectives – editing practice –Longer Documents: writing a proposal & Reports, Agenda – Minutes – Circular

TOTAL: 30 HOURS

ADDITIONAL EXPERIMENTS:

1. Body Language: Kinesics, Proxemics, Para linguistic, Nuances of Speech Delivery
2. Personality Development: Building self esteem
3. Team work

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1: To enable students to get International recognition for work and study.

CO2: To use English confidently in the International business environments.

CO3: To be able to take part in business discussion, read company literature, write formal and informal business correspondences and listen and understand business conversations

REFERENCES:

1. Guy Brook-Hart, "BEC VANTAGE: BUSINESS BENCHMARK Upper-Intermediate – Student's Book", 1st Edition, Cambridge University Press, New Delhi, 2006.
2. Cambridge Examinations Publishing, "Cambridge BEC VANTAGE – Self-study Edition", Cambridge University Press, UK, 2005.
3. Swets, Paul. W. 1983. The Art of Talking So That People Will Listen: Getting
4. The Process of Writing: Planning and Research, Writing, Drafting and Revising