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

B.E. Computer Science and Engineering

Full Time Curriculum and Syllabus

Second Year – Third Semester

Course Code	Course Name	L	T	P	C	Maximum Marks		
						CA	ES	Total
Theory Course								
1701MA301	Engineering Mathematics III	3	2	0	4	40	60	100
1702CS301	Data Structures	3	2	0	4	40	60	100
1702CS302	Computer Organization and Architecture	3	0	0	3	40	60	100
1702CS303	Software Engineering	3	0	0	3	40	60	100
1702CS304	Digital Systems	3	0	2	4	50	50	100
1702CS305	Operating Systems	3	0	0	3	40	60	100
Laboratory Course								
1702CS351	Data Structures Lab	0	0	2	1	50	50	100
1702CS352	Operating Systems Lab	0	0	2	1	50	50	100
1704GE351	Life Skills: Business English	0	0	2	0	100	0	100

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

1701MA301	ENGINEERING MATHEMATICS III (Common to B.E - Civil, CSE, EEE, Mech B.Tech- IT Degree Programmes)	L	T	P	C
		3	2	0	4

PREREQUISITE :

1. Engineering Mathematics I
2. Engineering Mathematics II

COURSE OBJECTIVES:

1. To introduce Fourier series analysis and applications in Engineering, apart from its use in solving boundary value problems.
2. To acquaint the student with Fourier transform techniques used in wide variety of situations.
3. To introduce the effective mathematical tools for the solutions of partial differential equations that model several physical processes and to develop Z transform techniques for discrete time systems.

UNIT I FOURIER SERIES **12 Hours**

Dirichlet's conditions – General Fourier series – Odd and even functions – Half range sine series – Half range cosine series – Parseval's identity – Harmonic analysis – Simple Applications

UNIT II FOURIER TRANSFORMS **12 Hours**

Statement of Fourier integral theorem – Fourier transform pair – Fourier sine and cosine transforms – Properties – Transforms of simple functions – Convolution theorem – Parseval's identity

UNIT III PARTIAL DIFFERENTIAL EQUATIONS **12 Hours**

Formation of partial differential equations – Singular integrals — Solutions of standard types of first order partial differential equations – Lagrange's linear equation — Linear partial differential equations of second order with constant coefficients of homogeneous type- Applications

UNIT IV APPLICATIONS OF PARTIAL DIFFERENTIAL EQUATIONS **12 Hours**

Classification of PDE – Solutions of one dimensional wave equation – One dimensional equation of heat conduction – Steady state solution of two dimensional equation of heat conduction.

UNIT V Z – TRANSFORMS AND DIFFERENCE EQUATIONS **12 Hours**

Z - transforms – Elementary properties – Inverse Z – transform (using partial fraction and residues) – Convolution theorem – Formation of difference equations – Solution of difference equations using Z – transform.

TOTAL: 60 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

1. Linear Algebra
2. Numerical Solution of non-homogeneous partial differential equations

COURSE OUTCOMES:

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

CO1 Use Fourier series analysis which is central to many applications in engineering
CO2 Apply Fourier transform techniques used in wide variety of situations
CO3 Compute the solution of partial differential equations
CO4 Solve boundary value problem using partial differential equation
CO5 Apply Z transform techniques for discrete time systems

REFERENCES:

1. Veerarajan. T., "Transforms and Partial Differential Equations", Second reprint, Tata McGraw Hill Education Pvt. Ltd., New Delhi, 2012
2. Grewal. B.S., "Higher Engineering Mathematics", 42nd Edition, Khanna Publishers, Delhi, 2012.
3. Bali.N.P and Manish Goyal, "A Textbook of Engineering Mathematics", 7th Edition, Laxmi Publications Pvt Ltd , 2007
4. Ramana.B.V., "Higher Engineering Mathematics", Tata Mc-GrawHill Publishing Company Limited, New Delhi, 2008.
5. Narayanan.S., Manicavachagom Pillay.T.K and Ramanaiah.G "Advanced Mathematics for Engineering Students" Vol. II & III, S.Viswanathan Publishers Pvt Ltd. 1998.
6. www.nptelvideos.in/2012/11/mathematics-iii.html

1702CS301	DATA STRUCTURES	L	T	P	C
		3	2	0	4

PREREQUISITE :

Programming in C.

COURSE OBJECTIVES:

1. Be exposed to the concepts of ADTs
2. Learn linear data structures – list, stack, and queue.
3. Be exposed to sorting, searching, hashing algorithms
4. Learn to apply Tree and Graph structures

UNIT I LINEAR DATA STRUCTURES – LIST 12 Hours

Introduction, Basic terminology Data structures - Data structure operations - Abstract Data Types (ADTs) – List ADT – array-based implementation – linked list implementation —singly linked lists- circularly linked lists- doubly-linked lists – applications of lists –Polynomial Manipulation – All operation (Insertion, Deletion, Merge, Traversal)

UNIT II LINEAR DATA STRUCTURES – STACK AND QUEUE 12 Hours

STACK: Array implementation, Linked list implementation, Applications of stack: Infix to Postfix, Evaluation of Postfix, Balancing symbols, Nested function calls, Recursion, Towers of Hanoi. QUEUE: Array implementation, Linked List implementation, Circular Queue.

UNIT III SORTING, SEARCHING AND HASH TECHNIQUES 12 Hours

Sorting algorithms: Insertion sort - Selection sort - Shell sort - Bubble sort - Quick sort - Merge sort - Radix sort – Searching: Linear search –Binary Search Hashing: Hash Functions – Separate Chaining – Open Addressing – Rehashing – Extendible Hashing

UNIT IV NON LINEAR DATA STRUCTURES – TREES 12 Hours

General trees, Terminology, Representation of trees, Tree traversal- Binary tree, Representation, Expression tree, Binary tree traversal, Binary Search Tree: Construction, Searching, Insertion, Deletion, AVL trees: Rotation, Insertion, Deletion, B-Trees, Splay trees, Red-Black Trees.

UNIT V NON LINEAR DATA STRUCTURES – GRAPHS 12 Hours

Representation of Graphs – Breadth-first search – Depth-first search – Topological sort – Minimum Spanning Trees – Kruskal's and Prim's algorithm – Shortest path algorithm – Dijkstra's algorithm – Bellman-Ford algorithm – Floyd - Warshall algorithm.

TOTAL: 60 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

1. Applications of queue: Priority queue, Double ended queue.
2. Threaded Binary Tree

COURSE OUTCOMES:

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

CO1 Implement abstract data types for linear data structures
CO2 Apply the different linear data structures to problem solutions.
CO3 Critically analyze the various algorithms
CO4 Have a comprehensive knowledge of Trees and their implementations
CO5 Learn advanced data structures like Graphs and their implementation

REFERENCES:

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", 2nd Edition, Pearson Education, 2011
2. Seymour Lipschutz, "Data Structures with C", McGraw Hill Education, Special Indian Edition, 2014.
3. A.V.Aho, J.E Hopcroft and J.D.Ullman, "Data structures and Algorithms", Pearson Education, First Edition Reprint 2003.
4. R.F.Gilberg, B.A.Forouzan, "Data Structures", Second Edition, Thomson India Edition, 2005.
5. ReemaThareja, "Data Structures Using C", Oxford Higher Education , First Edition, 2011.
6. http://nptel.ac.in/courses/106102064/1

1702CS302	COMPUTER ORGANIZATION AND ARCHITECTURE	L	T	P	C
		3	0	0	3

PREREQUISITE :

Programming in C

COURSE OBJECTIVES:

1. To make students understand the basic structure and operation of digital computer.
2. To study the concepts of pipelining.
3. To expose the students to the concept of parallelism
4. To familiarize the students with hierarchical memory system including cache Memories and virtual memory.

UNIT I STRUCTURE OF COMPUTERS & MACHINE INSTRUCTION 9 Hours

Functional units - Basic operational concepts - Bus structures - Software - performance – Technology– Instruction and instruction sequencing – Addressing modes – operations and operands-Basic I/O operations. ALU design – Fixed point and floating point operations

UNIT II BASIC PROCESSING UNIT 9 Hours

Fundamental concepts – Execution of a complete instruction – Multiple bus organization – Hardwired control – Micro programmed control – Nano programming.

UNIT III PIPELINING 9 Hours

Basic concepts – Data hazards – Instruction hazards – Influence on instruction sets –Data path and control considerations – Performance considerations – Exception handling.

UNIT IV PARALLELISM 9 Hours

Instruction-level-parallelism – Parallel processing challenges – Flynn's classification – Hardware multithreading – Multicore processors

UNIT V MEMORY AND I/O SYSTEMS 9 Hours

Memory hierarchy - Memory technologies – Cache basics – Measuring and improving cache performance - Virtual memory - Input/output system, programmed I/O, DMA and interrupts, I/O processors.

TOTAL: 45 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

ALU operations-MIPS-VLIW-How the processors are made from silicon mud-Creating Data path

COURSE OUTCOMES:

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

CO1 Understand basic operations and instructions
CO2 Design arithmetic and logic unit.
CO3 Design and analyze pipelined control units
CO4 Understand parallel processing architectures.
CO5 Evaluate performance of memory systems.

REFERENCES:

1. William Stallings “Computer Organization and Architecture” , Seventh Edition Reprint, Pearson Education, 2016
2. Vincent P. Heuring, Harry F. Jordan, “Computer System Architecture”, Second Edition, Pearson Education, 2005.
3. Govindarajalu, “Computer Architecture and Organization, Design Principles and Applications", first edition, Tata McGraw Hill, New Delhi, 2012.
4. V.P. Heuring, H.F. Jordan, “Computer Systems Design and Architecture”, 2nd Edition, Pearson Education, 2012.
5. https://onlinecourses.nptel.ac.in/noc18_cs29/preview

1702CS303	SOFTWARE ENGINEERING	L	T	P	C
		3	0	0	3

PREREQUISITE :

Programming in C

COURSE OBJECTIVES:

1. To help the students in understanding the basic theory of software engineering and to apply these basic theoretical principles to a software project development.
2. To guide students to develop skills that will enable them to construct software of high quality, software that is reliable and that is reasonably easy to understand, modify and maintain.
3. To provide an understanding of why these skills are important.

UNIT I SOFTWARE ENGINEERING CONCEPTS 9 Hours

Software Engineering introduction- Project management concepts - Software engineering paradigms – Generic process models, water fall life cycle model -prototype model - RAD model - spiral model - incremental model – Understanding requirements.

UNIT II MANAGING SOFTWARE PROJECTS 9 Hours

Metrics : Metrics in process and project domains - Software measurement - Metrics for software Quality - Integrating metrics in a software engineering process - Estimation , Scheduling – Risk Management – Review Techniques - Software quality assurance.

UNIT III DESIGN CONCEPTS 9 Hours

Design Process - Design Principles - Design Concepts - Software architecture – Architectural style, design and Mapping - user interface design.

UNIT IV SOFTWARE TESTING AND DEBUGGING 9 Hours

Testing Fundamentals and strategies - White-box and Black box testing - Basis path testing - dataflow testing - testing for special environments - Unit testing, - Integration testing – validation testing - system testing – debugging - software maintenance – software configuration management

UNIT V ADVANCED CONCEPTS 9 Hours

Computer Aided Software Engineering - Clean room software engineering – Reengineering - Reverse Engineering.

TOTAL: 45 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

1. Version management
2. ISO 9000 Quality Standards

COURSE OUTCOMES:

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

CO1 Build an appropriate process model for a given project

CO2 Analyze the principles at various phases of software development

CO3 Translate specifications into design and identify the components to build the architecture for a given problem, all using an appropriate software engineering methodology

CO4 Define a Project management plan and tabulate appropriate testing plans at different levels during the development of the software

CO5 Understand the software project estimation models and estimate the work to be done and resources required and the schedule for a software project

REFERENCES:

1. Roger S. Pressman, Software Engineering: A Practitioner's Approach, Mc-Graw Hill, 7th Edition, 2010.
2. Ian Somerville, Software Engineering,, Addison-Wesley, 8th edition, 2006.
3. Steve McConnell, Code Complete, Second Edition, Microsoft Press.
4. Richard E. Fairley, Software Engineering Concepts, McGraw- Hill, 1985

1702CS304	DIGITAL SYSTEMS	L	T	P	C
		3	0	0	3

PREREQUISITE :

Basic Electrical and Electronics Engineering

COURSE OBJECTIVES:

1. To train the students in basics of digital functions
2. To impart the students in the designing ability of combinational and sequential circuits
3. To educate the students about different types of memory and programmable devices
4. To teach the students about software skill in VHDL/Verilog HDL

UNIT I BOOLEAN ALGEBRA AND LOGIC GATES 9 Hours

Boolean Algebra: Boolean postulates and laws – De-Morgan's theorem - principle of duality - boolean expression - minimization of boolean expressions — minterm – maxterm - Sum of Products (SOP) – Product of Sums (POS) – Karnaugh map minimization – Quine - Mc Cluskey method of minimization.

Logic Gates: AND, OR, NOT, NAND, NOR, Exclusive–OR and Exclusive–NOR Implementations of logic functions using gates, NAND–NOR implementations – multi level gate implementations - multi output gate implementations. TTL and CMOS Logic and their characteristics – tristate gates

UNIT II COMBINATIONAL LOGICS 9 Hours

Introduction - design procedure – half adder – full adder – half subtractor – full subtractor – parallel binary adder, parallel binary subtractor – fast adder - carry look ahead adder – serial adder/subtractor - BCD adder – binary multiplier – binary divider - multiplexer/ demultiplexer – decoder - encoder – parity checker – parity generators – code converters - Magnitude Comparator – Seven segment display

UNIT III SYNCHRONOUS SEQUENTIAL LOGICS 9 Hours

Latches, Flip-flops - SR, JK, D, T, and Master-Slave – Characteristic table and equation –application table – Edge triggering – Level Triggering – Realization of one flip flop using other flip flops – serial adder/subtractor- synchronous counters – synchronous up/down counters – programmable counters – design of synchronous counters: state diagram- state table –state minimization –state assignment - excitation table and maps-circuit implementation - modulo–n counter, Registers – shift registers - universal shift registers - design of synchronous sequential circuits using VERILOG

UNIT IV ASYNCHRONOUS SEQUENTIAL LOGICS 9 Hours

Design of fundamental mode and pulse mode circuits – asynchronous ripple or serial counter – asynchronous up/down counter - state machines – problems in asynchronous circuits – static and dynamic hazards - design of hazard free switching circuits. design of asynchronous sequential circuits using VERILOG

UNIT V MEMORY AND PROGRAMMABLE LOGIC DEVICES 9 Hours

Classification of memories – ROM - ROM organization - PROM – EPROM – EEPROM –EAPROM, RAM – RAM organization – write operation – read operation – memory cycle - timing wave forms – memory decoding – memory expansion – static RAM cell- bipolar RAM cell – MOSFET RAM cell – dynamic RAM cell –programmable logic devices – Programmable Logic Array (PLA) - Programmable Array Logic (PAL) – Field Programmable Gate Arrays (FPGA) - Implementation of combinational logic circuits using ROM, PLA, PAL

TOTAL: 45 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

Modern Digital Design, Combinational Logic using VHDL Gate Models

COURSE OUTCOMES:

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

CO1 Use different methods which are used to simplify the Boolean functions
CO2 Demonstrate different types of combinational circuits to satisfy the user requirements
CO3 Implement various synchronous sequential circuits
CO4 Practice several types of asynchronous counters
CO5 Explain the basics of memory and programmable logic devices
CO6 Discuss the HDL Program for combinational and sequential circuits

REFERENCES:

1. John F.Wakerly, "Digital Design", Fourth Edition, Pearson/PHI, 2008
2. John.M Yarbrough, "Digital Logic Applications and Design", Thomson Learning, 2006
3. Charles H.Roth. "Fundamentals of Logic Design", 6th Edition, Thomson Learning, 2013
4. Donald P.Leach and Albert Paul Malvino, "Digital Principles and Applications", 6th Edition, TMH,2006
5. Thomas L. Floyd, "Digital Fundamentals", 10th Edition, Pearson Education Inc, 2011
6. Donald D.Givone, "Digital Principles and Design", TMH, 2003

1702CS305	OPERATING SYSTEMS	L	T	P	C
		3	0	0	3

PREREQUISITE :

Programming in C

COURSE OBJECTIVES:

1. Study the basic concepts and functions of operating systems.
2. Understand the structure and functions of OS.
3. Learn about Processes, Threads and Scheduling algorithms
4. Understand the principles of concurrency and Deadlocks.
5. Learn various memory management schemes
6. Study I/O management and File systems.

UNIT I INTRODUCTION **5 Hours**

Introduction- Operating System Structure – Operating System Operations – Process Management – Memory Management – Storage Management – Protection and Security – Distributed Systems –Computing Environments – System Structures: Operating System Services – User Operating System Interface – System Calls – Types of System Calls – System Programs. OS Generation and System Boot.

UNIT II PROCESS MANAGEMENT **12 Hours**

Processes-Process Concept, Process Scheduling, Operations on Processes, Inter process Communication; Threads- Overview, Multicore Programming, Multithreading Models; Windows 7 -Thread and SMP Management. Process Synchronization - Critical Section Problem, Mutex Locks, Semophores, Monitors; CPU Scheduling and Deadlocks. Deadlock Characterization – Methods for handling Deadlocks -Deadlock Prevention – Deadlock avoidance – Deadlock detection – Recovery from Deadlocks

UNIT III MEMORY MANAGEMENT **10 Hours**

Memory Management: Background – Swapping – Contiguous memory allocation –Paging – Segmentation – Segmentation with paging. Virtual Memory: Background –Demand paging – Process creation – Page replacement – Allocation of frames –Thrashing. Case Study: Memory management in Linux

UNIT IV STORAGE MANAGEMENT **9 Hours**

File System : File concept – Access methods – Directory structure – File system mounting – Protection. File-System Implementation : Directory implementation – Allocation methods – Free-space management – efficiency and performance – recovery. Case studies: File system in Linux – File system in Windows XP

UNIT V I/O SYSTEMS **9 Hours**

I/O Systems – I/O Hardware – Application I/O interface – kernel I/O subsystem –streams – performance. Mass-Storage Structure: Disk scheduling – Disk management –Swap-space management – RAID – disk attachment – stable storage – tertiary storage. Case study: I/O in Linux.

TOTAL: 45 HOURS

FURTHER READING / CONTENT BEYOND SYLLABUS / SEMINAR :

Linux System- Basic Concepts; System Administration-Requirements for Linux System Administrator-Setting up a LINUX Multifunction Server, Domain Name System, Setting Up Local Network Services; Virtualization- Basic Concepts, Setting Up Xen, VMware on Linux Host and Adding Guest OS.

COURSE OUTCOMES:

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

CO1 Understand Operating System Structure, Operations and Services& Illustrate the operating system concepts and its functionalities.

CO2 Understand the Process Concept, Multithreaded Programming, Process Scheduling and Synchronization

CO3 Apply the Concepts of Virtual Memory Management and File Systems

CO4 Analyze the Secondary Storage and I/O Systems

CO5 Evaluate the different Protection and Security Mechanisms for Operating System

REFERENCES:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, 9th Edition, John Wiley and Sons Inc., 2012.
2. Andrew S. Tanenbaum, “Modern Operating Systems”, Fourth Edition Prentice Hall of India Pvt. Ltd, 2015 .
3. Harvey M. Deitel, “Operating Systems”, Pearson Education Pvt. Ltd, Third Edition, 2004.
4. William Stallings, “Operating System”, Pearson Education, Sixth Edition, 2012.
5. Nptel reference: http://nptel.ac.in/courses/106106144/

1702CS351	DATA STRUCTURES LAB	L	T	P	C
		0	0	2	1

PREREQUISITE :

Programming in C

COURSE OBJECTIVES:

1. Be exposed to implementing abstract data types
2. Learn to implement sorting and searching algorithms.
3. Getting exposure in implementing the different data structures

LIST OF EXPERIMENTS:

1. Representation of records using Structures in C – Creation of Linked List – Manipulation of records in a Linked List
2. Operations on a Stack and Queue – infix to postfix – simple expression evaluation using stacks
3. Linked Stack Implementation – Linked Queue Implementation
4. Implementation of Sorting algorithms
5. Implementation of Linear search and Binary Search.
6. Applications of Stack and Queue
7. Binary Search Tree
8. Tree traversal Techniques
9. Minimum Spanning Trees.
10. Shortest Path Algorithms - Dijkstra's algorithm .

TOTAL: 45 HOURS

ADDITIONAL EXPERIMENTS / INNOVATIVE EXPERIMENTS

1. Program to construct an expression tree for a given tree
2. Implementation of Bellman-Ford algorithm and Floyd - Warshall algorithm.

COURSE OUTCOMES:

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

CO1 Design and implement C programs for implementing stacks, queues, linked lists

CO2 Develop searching and sorting programs.

CO3 Apply the different data structures for implementing solutions to practical problems.

CO4 Develop recursive programs using trees and graphs

REFERENCES:

1. www.cs.cf.ac.uk/Dave/C/
2. http://www.lysator.liu.se/c/bwk-tutor.html
3. http://en.wikibooks.org/wiki/Data_Structures/Introduction
4. http://www.eskimo.com/~scs/cclass/notes/top.html

1702CS352	OPERATING SYSTEMS LAB	L	T	P	C
		0	0	2	1

PREREQUISITE :

Programming in C

COURSE OBJECTIVES:

1. To gain a complete knowledge about UNIX commands and shell programming
2. To obtain an overview of distributed operating systems and the related topics of inter process communication models (message passing, remote procedure call, distributed object computing, and shared memory)
3. To know the concepts of process management and synchronization
4. To know the concept of memory management such as best fit, worst fit and so on

LIST OF EXPERIMENTS:

1. Study of basic Commands in Unix Operating System
2. Shell programming using control statements
3. Shell programming using loops, patterns, expansions and substitutions
4. Write programs using the following system calls (fork, exec, getpid, exit, wait, close, stat, opendir, readdir).
5. Write programs using the I/O system calls (open, read, write, etc).
6. Simulation of Unix commands.
7. Implementation of CPU Scheduling Algorithms(FCFS, SJF, RR, Priorty).
8. Implementation of Page Replacement Algorithms (LRU, OPT, FIFO).
9. Implementation of memory allocation algorithms (First Fit, Best Fit, Worst Fit)
10. Implement the Producer – Consumer problem using semaphores.
11. Simulation of Shared Memory Concept.
12. Implementation of bankers Algorithm.
13. Implement Paging Technique of memory management.
14. Implementation Disk Scheduling Algorithms
15. Study of Linux OS, Android OS.

TOTAL: 45 HOURS

ADDITIONAL EXPERIMENTS / INNOVATIVE EXPERIMENTS

1. Implement some memory management schemes
2. Application Oriented Experiments
3. Mini Project

COURSE OUTCOMES:

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

CO1 The student will be familiar with the language and terms of the UNIX/LINUX operating system

CO2 The student will be able to delineate the commands and procedures needed to carry out basic operations on the UNIX/LINUX operating system

CO3 Students can design, develop and implement a software solution to a given problem which employs operating systems tools

REFERENCES:

1. http://www.ee.surrey.ac.uk/Teaching/Unix/unixintro.html
2. https://kb.iu.edu/d/afsk
3. http://www.ch.embnet.org/CoursEMBnet/Pages05/slides/Unix05.pdf
4. http://www.ee.surrey.ac.uk/Teaching/Unix/
5. http://www.comptechdoc.org/os/linux/usersguide/linux_ugshellpro.html
6. http://www.cs.jhu.edu/~yairamir/cs418/os4/sld025.html

1704GE351	LIFE SKILLS : SOFT SKILLS	L	T	P	C
	(Common to all B.E / B.Tech Degree Programmes)	0	0	2	0

PREREQUISITE :

1. Technical English
2. Communicative English

COURSE OBJECTIVES:

1. To develop the students basic soft skills and enable them to get a job.
2. To develop the students' interpersonal skills and to enable them to respond effectively.
3. To develop the students selling skills and to enable them to apply in their interview process.
4. To develop the students' Corporate Etiquettes and enable them to respond effectively.
5. To develop the students' learning by practice of giving different situations.

UNIT I INTRODUCTION TO SOFT SKILLS 6 Hours

Soft Skills an Overview - Basics of Communication – Body Language – Positive attitude –Improving Perception and forming values – Communicating with others.

UNIT II TEAM Vs TRUST 6 Hours

Interpersonal skills – Understanding others – Art of Listening - Group Dynamics – Networking - Individual and group presentations - Group interactions – Improved work Relationship .

UNIT III SELLING ONESELF 6 Hours

How to brand oneself – social media – job hunting – Resume writing – Group Discussion – Mock G.D - .Interview skills – Mock Interview

UNIT IV CORPORATE ETIQUETTES 6 Hours

What is Etiquette – Key Factors – Greetings – Meeting etiquettes – Telephone etiquettes – email etiquettes – Dining etiquettes – Dressing etiquettes – Rest room etiquettes – Life etiquettes.

UNIT V LEARNING BY PRACTICE 6 Hours

1. My family. Myself. 2. Meeting people. Making Contacts. 3. A city. Getting about town. 4. Our flat. Home life.5. Travelling. Going abroad. 6. Going through Customs. 7. At a hotel. 8. Shopping. 9. Eating out. 10. Making a phone call. 11. A modern office.12 Discussing business.

TOTAL: 30 HOURS

ASSSESSMENT PATTERN

1. Two assignments (2 x 25 marks = 50 marks)
2. Pragmatic assessment (50 marks)

COURSE OUTCOMES:

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

CO1 Communicate effectively in their business environment.
CO2 Improve their interpersonal skills which are mandatory in a corporate world.
CO3 Brand themselves to acquire a job.
CO4 Involve in corporate etiquettes.
CO5 Survive in the different situations.

REFERENCES:

1. Dr.K.Alex, 'Soft Skills' Third Edition, S.Chand & Publishing Pvt Limited, 2009
2. Aruna Koneru, 'Professional Communication' Second Edition, Tata McGraw-Hill Education, 2008
3. D.K.Sarma, 'You & Your Career' First Edition, Wheeler Publishing & Co Ltd, 1999
4. Shiv Khera 'You Can Win' Third Edition, Mac Millan Publisher India Pvt Limited, 2005