B.Tech.

in

COMPUTER SCIENCE & ENGINEERING (DATA SCIENCE)
[CSE (DS)]

Scheme of Instruction, Examination and Syllabi

of

III and IV Semesters

Academic Year: 2022-23

MAHATMA GANDHI INSTITUTE OF TECHNOLOGY

(An Autonomous Institution)
Affiliated to JNTUH; Accredited by NAAC with 'A' Grade; 6 U.G. Programs
Accredited by NBA
Kokapet (Village), Gandipet (Mandal), Hyderabad-500075, Telangana
email: principal@mgit.ac.in Website: www.mgit.ac.in Ph: 040-24193057 / 067

MAHATMA GANDHI INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech. in Computer Science & Engineering (Data Science)

Scheme of Instruction and Examination

III Semester

S.No.	Course Code	Course Title	Instructions			Examination			Credits
			Hours per work			Max. Marks		Duration of SEE in Hours	
			L	T	P/D	CIE	SEE		
1	MA305BS	Mathematical and Statistical Foundations	3	0	0	30	70	3	3
2	CS302PC	Fundamentals of Data Structures	3	1	0	30	70	3	4
3	CS305PC	Python Programming	3	0	0	30	70	3	3
4	CS306PC	Discrete Mathematics	3	0	0	30	70	3	3
5	CS308PC	Computer Organization and Architecture	3	0	0	30	70	3	3
6	CS351PC	Data Structures Lab	0	0	3	30	70	3	1.5
7	CS355PC	Python Programming Lab	0	0	3	30	70	3	1.5
8	CS357PC	Introduction to IT Workshop	0	0	2	30	70	3	1
9	EN351HS	Finishing School-I	0	0	2	30	70	3	1
10	MC351HS	Gender Sensitization Lab	0	0	2	30	70	3	0
Total Hours/Marks/Credits			**15**	**1**	**12**	**300**	**700**	**--**	**21**

L: Lecture **T**: Tutorial **D**: Drawing **P**: Practical
CIE - Continuous Internal Evaluation **SEE -** Semester End Examination

L	T	P	C
3	0	0	3

III Semester Syllabus
MA305BS: Mathematical and Statistical Foundations

[Common to CSE (AI&ML) & CSE (DS)]

Course Objectives

To Learn

- To understand the basic Number Theory concepts useful for computer organization and security, coding and cryptography.
- To understand simple linear regression, the linear and nonlinear curve fitting of the given data.
- To understand the theory of probability distributions.
- To gain the knowledge of the sampling theory, testing of hypothesis and making inferences.
- To understand the stochastic process and Markov chains.

Course Outcomes

After learning the contents of this course, the student must be able to

- Apply the number theory concepts to cryptography domain.
- Apply the concepts of probability and distributions to some case studies.
- Correlate the material of one unit to the material in other units.
- Resolve the potential misconceptions and hazards in each topic of study.

Unit – I: Greatest Common Divisors and Prime Factorization

Greatest common divisors, The Euclidean algorithm, the fundamental theorem of arithmetic, Factorization of integers and the Fermat numbers.

Congruences: Introduction to congruences, Linear congruences, The Chinese remainder theorem, Systems of linear congruences.

Unit – II: Simple Linear Regression

Introduction to Linear Regression, The Simple Linear Regression Model, Least Squares and the Fitted Model, Properties of the Least Squares Estimators, Inferences Concerning the Regression Coefficients, Prediction, Simple Linear Regression Case Study.

Unit – III: Random Variables and Probability Distributions

Concept of a Random Variable, Discrete Probability Distributions, Continuous Probability Distributions, Statistical Independence.

Discrete Probability Distributions: Binomial Distribution, Poisson distribution.

Continuous Probability Distributions: Normal Distribution, Areas under the Normal Curve, Applications of the Normal Distribution, Normal Approximation to the Binomial.

Fundamental Sampling Distributions: Random Sampling, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem, Sampling Distribution of S2, t–Distribution, F- Distribution.

Unit – IV: Estimation & Tests of Hypotheses

Introduction, Statistical Inference, Classical Methods of Estimation, Estimating the Mean, Standard Error of a Point Estimate, Prediction Intervals, Tolerance Limits, Estimating the Variance, Estimating single mean, difference between two Means, single proportion, difference between two proportions.

Unit – V: Stochastic Process and Markov Chains

Introduction to Stochastic process- Markov process. Transition Probability, Transition Probability Matrix, First order and Higher order Markov process, n-step transition probabilities, Markov chain, Steady state condition, Markov analysis.

Suggested Readings:

1. Kenneth H. Rosen, Elementary number theory & its applications, sixth edition, Addison-Wesley, ISBN 978 0-321-50031-1
2. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Probability & Statistics for Engineers & Scientists, 9th Ed. Pearson Publishers.
3. S. D. Sharma, Operations Research, Kedarnath and Ramnath Publishers, Meerut, Delhi.

Reference Books:

1. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications.
2. T.T. Soong, Fundamentals of Probability and Statistics for Engineers, John Wiley & Sons Ltd, 2004.
3. Sheldon M Ross, Probability and statistics for Engineers and scientists, Academic Press.

L	T	P	C
3	1	0	4

III Semester Syllabus
CS302PC: Fundamentals of Data Structures
[Common to CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- Exploring linear data structures such as stacks, queues and linked lists.
- Introduces a variety of data structures such as hash tables and dictionaries. Exploring non-linear data structures such as trees and graphs.
- Introduces a variety of data structures such as search trees, heaps and tries.
- Introduces sorting and pattern matching algorithms

Course Outcomes

- Ability to select the data structures that efficiently model the information in a problem.
- Ability to assess efficiency trade-offs among different data structure implementations or combinations.
- Implement and know the application of algorithms for sorting and pattern matching.
- Design programs using a variety of data structures, including hash tables, binary and general tree structures, search trees, tries, heaps, graphs, and AVL-trees.
- To impart knowledge on various string-matching algorithms and implement them.

Unit – I

Introduction to Data Structures: abstract data types, Linear list – singly linked list implementation, insertion, deletion and searching operations on linear list, Stacks-Operations, array and linked representations of stacks, stack applications, Queues-operations, array and linked representations.

Unit – II

Dictionaries: linear list representation, skip list representation, operations **-** insertion, deletion and searching.

Hash Table Representation: hash functions, collision resolution-separate chaining, open addressing**-** linear probing, quadratic probing, double hashing, rehashing, extendible hashing.

Unit – III

Search Trees: Binary Search Trees, Definition, Implementation, Operations- Searching, Insertion and Deletion, AVL Trees, Definition, Height of an AVL Tree, Operations – Insertion, Deletion and Searching, Red –Black, Splay Trees.

Unit – IV

Graphs: Graph Implementation Methods. Graph Traversal Methods.

Sorting: Heap Sort, External Sorting- Model for external sorting, Merge Sort.

Unit – V

Pattern Matching and Tries: Pattern matching algorithms-Brute force, the Boyer –Moore algorithm, the Knuth-Morris-Pratt algorithm, Standard Tries, Compressed Tries, Suffix tries.

Suggested Readings:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C++, Varsha H. Patil, OXFORD University Press.

Reference Books:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B.A. Forouzan, Cengage Learning.
2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson Education.

L	T	P	C
3	0	0	3

III Semester Syllabus
CS305PC: Python Programming

[Common to CSE (AI&ML) & CSE (DS)]

CourseObjectives

- To learn about Python programming language syntax, semantics, and the runtime environment
- To be familiarized with universal computer programming concepts like data types, containers
- To be familiarized with general computer programming concepts like conditional execution, loops & functions
- To be familiarized with general coding techniques and object-oriented programming

Course Outcomes

- Develop essential programming skills in computer programming concepts like data types, containers.
- Apply the basics of programming in the Python language.
- Solve coding tasks related Exceptions and Functions.
- Solve coding tasks related to the fundamental notions and techniques used in object-oriented programming.
- Solve coding tasks related to GUI Programming.

Unit – I

Introduction: Introduction to Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, Operators. Type conversions, Expressions, More about Data Output.

Data Types and Expression: Strings Assignment, and Comment, Numeric Data Types and Character Sets, Using functions and Modules.

Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables.

Repetition Structures: Introduction, while loop, for loop, Input Validation Loops, Nested Loops.

Unit – II

Sequences: Introduction to Sequences: Strings, Lists, and Tuples, Mapping, Set Types and Dictionaries.

String: Strings and Operators, Built-in Methods and Special Features of String. Related Modules.

Lists: Operators, Built-in Functions, List Type Built in Methods, Special Features of List, Related Modules

Tuples: Built-in Functions, Tuple Type Built in Methods, Special Features of Tuples, Related Modules

Mapping and Set Type: Dictionaries, Dictionary Keys, Operators and Built-in Methods. Set type, Operator and Built in Methods. Related Modules.

Unit – III

Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Redefined Clean-up Actions.

Functions: What are Functions, Defining and Creating functions, Function Arguments: Formal and Variable length, Calling functions, Recursive Functions and Variable Scope.

Modules: Modules, Standard Modules, Importing Modules, Namespaces and Packages.

Unit – IV

File Operations: Reading config files in python, Writing log files in python, Understanding read functions, read(), readline() and readlines(), Understanding write functions, write() and writelines(), Manipulating file pointer using seek, Programming using file operations

Object Oriented Programming: Concept of class, object and instances, Constructor, class attributes and destructors, Real time use of class in live projects, Inheritance , overlapping and overloading operators, Adding and retrieving dynamic attributes of classes, Programming using OOPs support

Design with Classes: Objects and Classes, Data modelling Examples, Case Study An ATM, Structuring Classes with Inheritance and Polymorphism.

Unit – V

GUI Programming: Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs.

Suggested Readings:

1. Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage.
2. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.

Reference Books:

1. Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.
2. Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

e-Resources:

https://www.tutorialspoint.com/python3/python_tutorial.pdf

L	T	P	C
3	0	0	3

III Semester Syllabus
CS306PC: Discrete Mathematics

[Common to CSE, CSE (AI&ML) & CSE (DS)]

Course Objectives

- Introduces the elementary discrete mathematics for computer science and engineering.
- Topics include formal logic notation, methods of proof, induction, sets, relations, graph theory, permutations, and combinations, counting principles; recurrence relations and generating functions.

Course Outcomes

- Ability to understand logical inferences and first order logic.
- Ability to use sets, functions and relations.
- Ability to analyze and solve counting problems on finite and discrete structures.
- Ability to formulate recurrences relations to solve problems.
- Ability to apply graph theory in solving computing problems.

Unit-I

The Foundations: Logic and Proofs: Propositional Logic, Applications of Propositional Logic, Propositional Equivalence, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.

Unit-II

Basic Structures, Sets, Functions, Sequences, Sums, Matrices and Relations Sets, Functions, Sequences & Summations, Cardinality of Sets and Matrices Relations, Relations and Their Properties, n-ary Relations and Their Applications, Representing Relations, Closures of Relations, Equivalence Relations, Partial Orderings.

Unit-III

Induction and Recursion and Counting: Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness.

Counting: The Basics of Counting, the Pigeonhole Principle, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Generating Permutations and Combinations

Unit-IV

Discrete Probability and Advanced Counting Techniques: An Introduction to Discrete Probability, Probability Theory, Bayes’ Theorem, Expected Value and Variance. Advanced Counting Techniques: Recurrence Relations, Solving Linear Recurrence Relations. Divide-and-

Conquer Algorithms and Recurrence Relations, Generating Functions.

Unit – V

Graphs: Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring.
Trees: Introduction to Trees, Applications of Trees, Tree Traversal, Spanning Trees, Minimum Spanning Trees.

Suggested Readings:

1. Discrete Mathematics and its Applications with Combinatorics and Graph Theory- Kenneth HRosen, 7th Edition, TMH.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe L. Mott, Abraham Kandel, Teodore P. Baker, 2nd edition, Pearson Education.

References Books:

1. Discrete Mathematical Structures with Applications to Computer Science - J.P. Tremblay and R. Manohar, TMH,
2. Discrete Mathematics- Richard Johnson Baugh, 7th edition, Pearson Education.
3. Discrete Mathematics with Graph Theory- Edgar G. Goodaire, Michael M. Parmenter.
4. Discrete and Combinatorial Mathematics - an applied introduction: Ralph.P. Grimald, 5th edition, Pearson Education.

L	T	P	C
3	0	0	3

III Semester Syllabus
CS308PC: Computer Organization and Architecture
[Common to CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- The purpose of the course is to introduce principles of computer organization and the basic architectural concepts.
- It begins with basic organization, design, and programming of a simple digital computer and introduces simple register transfer language to specify various computer operations.
- Topics include computer arithmetic, instruction set design, micro programmed control unit, pipe lineing and vector processing, memory organization and I/O systems and multiprocessors.

Course Outcomes

- Understand the basics of instructions and their impact on processor design.
- Demonstrate an understanding of the design of the functional units of a digital computer system.
- Evaluate cost performance and design trade-offs in designing a memory unit.
- Design a pipe line for consistent execution of instructions with minimum hazards.
- Recognize and manipulate representations of numbers stored in digital computers.

Unit-I

Digital Computers: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Register Transfer Language and Micro operations: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.

Basic Computer Organization and Design: Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

Unit-II

Microprogrammed Control: Control memory, Address sequencing, micro program example, design of control unit.

Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

Unit-III

Data Representation: Data types, Complements, Fixed Point Representation, Floating Point

Representation.

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating–point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

Unit-IV

Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

Unit-V

Reduced Instruction Set Computer: CISC Characteristics, RISC Characteristics.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor.

Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Interprocessor arbitration, Interprocessor communication and synchronization, Cache Coherence.

Suggested Readings:

1. Computer System Architecture–M.Moris Mano, Third Edition, Pearson/PHI.

Reference Books:

1. Computer Organization–Car Hamacher, Zvonks Vranesic, SafeaZaky, Vth Edition, McGraw Hill.
2. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, PHI/Pearson.

L	T	P	C
0	0	3	1.5

III Semester Syllabus
CS351PC: Data Structures Lab
[Common to CSE, CSE (AI&ML) & CSE (DS)]

Course Objectives

- It covers various concepts of C programming language.
- It introduces searching and sorting algorithms.
- It provides an understanding of data structures such as stacks and queues.

Course Outcomes

- Ability to develop C programs using basic elements like control statements, arrays,
- Ability to develop C programs using functions, pointers and strings,
- Ability to develop C programs using data structures like stacks, queues and linked lists.
- Ability to Implement searching algorithms
- Ability to Implement sorting algorithms

List of Experiments:

1. Write a program to implement stack using the following data structures using arrays
   i) Stacks ii)Queues
2. Write a program to implement stack using the following data structures using linked list.
   i) Stacks ii)Queues
3. Write a program that uses functions to perform the following operations on singly linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Write a program that uses functions to perform the following operations on doubly linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
5. Write a program that uses functions to perform the following operations on Circular linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
6. Write a program to implement the Binary search tree traversal methods.
7. Write a program to implement the graph traversal methods.
   i) BFS ii) DFS
8. Write a program that use both recursive and non-recursive functions to perform the following searching operations for a Key value in a given list of integers:
   i) Linear search ii) Binary search
9. Write a program that implements the following sorting methods to sort a given list of integers in ascending order
   i) Bubble sort ii) Selection sort iii) Insertion sort

Reference Books:

1. Data Structures: A Pseudo code Approach with C, 2nd Edition, R.F.Gilberg and B.A.Forouzan, Cengage Learning.

L	T	P	C
0	0	3	1.5

III Semester Syllabus
CS355PC: Python Programming Lab

[Common to CSE (AI&ML) &CSE (DS)]

Course Objectives

- To acquire programming skills in core Python.
- To acquire Object Oriented Skills in Python
- To develop the skill of designing Graphical user Interfaces in Python

Course Outcomes

- Write, Test and Debug Python Programs
- Use Conditionals and Loops for Python Programs
- Use functions and represent Compound data using Lists, Tuples and Dictionaries
- Use various applications using python

List of Experiments:

1) Write a program that asks the user for a weight in kilograms and converts it to pounds. There are 2.2 pounds in a kilogram.
2) Write a program that asks the user to enter three numbers (use three separate input statements). Create variables called total and average that hold the sum and average of the three numbers and print out the values of total and average.
3) Write a program that uses a *for* loop to print the numbers 8, 11, 14, 17, 20, . . . , 83, 86, 89.
4) Write a program that asks the user for their name and how many times to print it. The program should print out the user's name the specified number of times.
5) Use a *for*loop to print a triangle like the one below. Allow the user to specify how high the triangle should be.

```
*
**
***
****
```

6) Generate a random number between 1 and 10. Ask the user to guess the number and print a message based on whether they get it right or not.
7) Write a program that asks the user for two numbers and prints *Close* if the numbers are within .001 of each other and not close otherwise.
8) Write a program that asks the user to enter a word and prints out whether that word contains any vowels.
9) Write a program that asks the user to enter two strings of the same length. The program should then check to see if the strings are of the same length. If they are not, the program should print an appropriate message and exit. If they are of the same length, the program

should alternate the characters of the two strings. For example, if the user enters *abcde*and*ABCDE* the program should print out *AaBbCcDdEe*.

10) Write a program that asks the user for a large integer and inserts commas into it according to the standard American convention for commas in large numbers. For instance, if the user enters 1000000, the output should be 1,000,000.
11) In algebraic expressions, the symbol for multiplication is often left out, as in 3x+4y or 3(x+5). Computers prefer those expressions to include the multiplication symbol, like 3*x+4*y or 3*(x+5). Write a program that asks the user for an algebraic expression and then inserts multiplication symbols where appropriate.
12) Write a program that generates a list of 20 random numbers between 1 and 100.
    (a) Print the list.
    (b) Print the average of the elements in the list.
    (c) Print the largest and smallest values in the list.
    (d) Print the second largest and second smallest entries in the list
    (e) Print how many even numbers are in the list.
13) Write a program that asks the user for an integer and creates a list that consists of the factors of that integer.
14) Write a program that generates 100 random integers that are either 0 or 1. Then find the longest run of zeros, the largest number of zeros in a row. For instance, the longest run of zeros in [1,0,1,1,0,0,0,0,1,0,0] is 4.
15) Write a program that removes any repeated items from a list so that each item appears at most once. For instance, the list [1,1,2,3,4,3,0,0] would become [1,2,3,4,0].
16) Write a program that asks the user to enter a length in feet. The program should then give the user the option to convert from feet into inches, yards, miles, millimeters, centimeters, meters, or kilometers. Say if the user enters a 1, then the program converts to inches, if they enter a 2, then the program converts to yards, etc. While this can be done with if statements, it is much shorter with lists and it is also easier to add new conversions if you use lists.
17) Write a function called *sum_digits*that is given an integer num and returns the sum of the digits of num.
18) Write a function called *first_diff*that is given two strings and returns the first location in which the strings differ. If the strings are identical, it should return -1.
19) Write a function called *number_of_factors*that takes an integer and returns how many factors the number has.
20) Write a function called *is_sorted*that is given a list and returns True if the list is sorted and False otherwise.
21) Write a program to implement various built in methods of tuples.
22) Write a function called root that is given a number x and an integer n and returns $x^{1/n}$. In the function definition, set the default value of n to 2.
23) Write a function called primes that is given a number n and returns a list of the first n primes. Let the default value of n be 100.
24) Write a function called merge that takes two already sorted lists of possibly different lengths, and merges them into a single sorted list.
    (a) Do this using the sort method. (b) Do this without using the sort method.

25) Write a program that asks the user for a word and finds all the smaller words that can be made from the letters of that word. The number of occurrences of a letter in a smaller word can't exceed the number of occurrences of the letter in the user's word.
26) Write a program that reads a file consisting of email addresses, each on its own line. Your program should print out a string consisting of those email addresses separated by semicolons.
27) Write a program that reads a list of temperatures from a file called *temps.txt*, converts those temperatures to Fahrenheit, and writes the results to a file called ftemps.txt.
28) Write a class called Product. The class should have fields called name, amount, and price, holding the product's name, the number of items of that product in stock, and the regular price of the product. There should be a method *get_price*that receives the number of items to be bought and returns a the cost of buying that many items, where the regular price is charged for orders of less than 10 items, a 10% discount is applied for orders of between 10 and 99 items, and a 20% discount is applied for orders of 100 or more items. There should also be a method called *make_purchase*that receives the number of items to be bought and decreases amount by that much.
29) Write a class called Time whose only field is a time in seconds. It should have a method called *convert_to_minutes*that returns a string of minutes and seconds formatted as in the following example: if seconds is 230, the method should return '5:50'. It should also have a method called *convert_to_hours*that returns a string of hours, minutes, and seconds formatted analogously to the previous method.
30) Write a class called Converter. The user will pass a length and a unit when declaring an object from the class—for example, c = Converter (9,'inches'). The possible units are inches, feet, yards, miles, kilometers, meters, centimeters, and millimeters. For each of these units there should be a method that returns the length converted into those units. For example, using the Converter object created above, the user could call c.feet() and should get 0.75 as the result.
31) Write a Python class to implement pow(x, n).
32) Write a Python class to reverse a string word by word.
33) Write a program that opens a file dialog that allows you to select a text file. The program then displays the contents of the file in a textbox.
34) Write a program to demonstrate Try/except/else.
35) Write a program to demonstrate try/finally and with/as.

Suggested Readings:

1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.

Reference Books:

1. Think Python, Allen Downey, Green Tea Press
2. Introduction to Python, Kenneth A. Lambert, Cengage
3. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
4. Learning Python, Mark Lutz, O'Reilly.

L	T	P	C
0	0	2	1

III Semester Syllabus
CS357PC: Introduction to IT Workshop
[Common to CSE (AI&ML) & CSE (DS)]

Course Objectives

The IT Workshop for engineers is a training lab course spread over 60 hours.

The modules include training on PC Hardware, Internet & World Wide Web and Productivity tools including Word, Excel, Power Point and Publisher.

- PC Hardware introduces the students to a personal computer and its basic peripherals.
- The process of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers. The students should work on working PC to disassemble and assemble to working condition and install Windows and Linux on the same PC. Students are suggested to work similar tasks in the Laptop scenario wherever possible.
- Hardware and software level troubleshooting process, tips and tricks would be covered.
- Internet & World Wide Web module introduces the different ways of hooking the PC on to the internet from home and workplace and effectively usage of the internet. Usage of web browsers, email, newsgroups and discussion forums would be covered. In addition, awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks would be introduced.
- Productivity tools module would enable the students in crafting professional word documents, excel spread sheets, power point presentations and personal web sites using the Microsoft suite of office tools and LaTeX.

Course Outcomes

- Understand and Identify the PC hardware and basic peripherals.
- Analyze and Practice of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers.
- Understand the Hardware and software level troubleshooting process.
- Explore the usage of Internet and WWW.
- Understand the need of documentation tools.
- Generate templates for generation report using Latex.

PC Hardware:

Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Task 2: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Task 3: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Task 4: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Task 5: Hardware Troubleshooting:

Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Task 6: Software Troubleshooting:

Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Internet & World Wide Web:

Task1: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Task 2: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Task 3: Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

Task 4: Cyber Hygiene: Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to first install an antivirus software, configure their personal firewall and windows update on their computer. Then they need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

LaTeX and WORD:

Task 1 – Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office 2007/ equivalent (FOSS) tool word: Importance of LaTeX and MS office 2007/ equivalent (FOSS) tool Word as word Processors, Details of the four tasks and features that would be covered in each, Using LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.

Task 2: Using LaTeX and Word to create project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Task 3: Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 4 : Creating a Newsletter : Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

MS Excel Orientation:

The mentor needs to tell the importance of MS office 2007/ equivalent (FOSS) tool Excel as a Spreadsheet tool, give the details of the four tasks and features that would be covered in each. Using Excel – Accessing, overview of toolbars, saving excel files, Using help and resources.

Task 1: Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, auto fill, Formatting Text.

Task 2 : Calculating GPA - .Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP .

Task 3: Performance Analysis - Features to be covered:- Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting.

MS Power Point:

Task 1:

Students will be working on basic power point utilities and tools which help them create basic power point presentation.

Topic covered during this week includes: - PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in PowerPoint.

Students will be given model power point presentation which needs to be replicated (exactly how it’s asked).

Task 2: Second week helps students in making their presentations interactive. Topic covered during this week includes: Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3:

Concentrating on the in and out of Microsoft power point and presentations. Helps them learn best practices in designing and preparing power point presentation.

Topic covered during this week includes: - Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), and Inserting – Background, textures, Design Templates, Hidden slides.

Reference Books:

1. Comdex Information Technology course tool kit Vikas Gupta, WILEY Dreamtech.
2. The Complete Computer upgrade and repair book, 3rd edition Cheryl A Schmidt, WILEY Dreamtech.
3. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
4. PC Hardware - A Handbook – Kate J. Chase PHI (Microsoft).
5. LaTeX Companion – Leslie Lamport, PHI/Pearson.
6. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme – CISCO Press, Pearson Education.
7. IT Essentials PC Hardware and Software Labs and Study Guide Third Edition by Patrick Regan – CISCO Press, Pearson Education.

L	T	P	C
0	0	2	1

III Semester Syllabus
EN351HS: Finishing School-I
(Common to all Branches)

Course Overview

In view of the growing importance of English as a tool for global Communication and the consequent emphasis on training students to acquire language skills, this syllabus has been designed to develop linguistic, communicative and critical thinking competencies of Engineering students.

Course Objectives

The main objective of this finishing school curriculum is to provide content for developing the LSRW skills of language learning and to facilitate proficiency in both receptive and productive skills, among students.

Methodology:

- Every Session will have activities on all the four skills-Listening, Speaking, Reading and Writing.
- To personalize the learning a variety of case studies and structured problem solving activities will be given to small groups and the teachers will facilitate peer reviews.
- Continuous grading, peer review and positive reinforcement will be emphasized
- Vocabulary exercises will also be a part of every session
- All sessions are designed to be student-centric and interactive.

Unit-I: Fundamentals of Communication

Unit Overview:

This is an introductory module that covers the fundamentals of communication. This module is intended to enable the students to communicate using greetings and small sentences/queries.

Learning Outcomes:

The students should be able to:

- Respond to questions
- Engage in informal conversations.
- Speak appropriately in formal situations
- Write formal and informal emails/letters

Competencies:

- Greeting appropriately
- Introducing themselves, a friend
- Situational Dialogue writing
- Responding to simple statements and questions both verbally and in writing
- Writing an email with appropriate salutation, subject lines, introduction and purpose of

mail.

- Using appropriate vocabulary for both formal and informal situations.
- JAM sessions.

Sessions:

1. Introduction to Formal and Informal Conversations (Listening Activity)
2. Informal Conversations
3. Informal Conversations - Writing
4. Formal Conversations
5. Formal Conversations - Writing
6. Grammar-Prepositions
7. Adjectives and Degrees of Comparison
8. Word formation: Prefixes and Suffixes

Unit–II: Rational Recap

Unit Overview:

The module enables the participants to organize their communication, structure their speaking and writing, explain their thoughts/ideas, and summarize the given information.

Learning Outcomes:

The students should be able to:

- Classify content and describe in a coherent form
- Recognize and list the key points in a topic/message/article.
- Compare and contrast using appropriate structure
- Explain cause and effect
- Use appropriate transitions in their presentations and written assignments

Competencies:

- Organizing the communication based on the context and audience
- Structuring the content based on the type of information.
- Explaining a technical/general topic in detail.
- Writing a detailed explanation/process
- Recapitulating

Sessions:

1. Introduction to Mind maps
2. Classification
3. Sequencing
4. Description and Enumeration

Unit-III: Narrations and Dialogues

Unit Overview:

The Module is intended to develop the desired level of language competence that enables them to narrate and participate in casual dialogues.

Learning Outcomes:

The students should be able to

- Narrate a message/story/incident, both verbally and in writing.

- Describe an event/a session/ a movie/ an object / image
- Understand Vocabulary in context

Competencies:

- Framing proper phrases and sentences to describe in context
- Reading Stories and articles and summarizing.
- Speaking fluently with clarity
- Listening for main ideas and reformulating information in his/her own words
- Drawing and write appropriate conclusions, post reading a passage.
- Speaking Reading and Writing descriptive sentences and paragraphs
- Using appropriate tenses, adjectives and adverbs in conversations and written tasks

Sessions:

Grammar: Verb, Tenses

1. Recalling and Paraphrasing
2. Describing Events
3. Describing Objects/ Places
4. Story Telling
5. Describing Hypothetical events

Unit-IV: Technical Expositions and Discussions

Unit Overview:

The module enables the students to build strategies for effective interaction and help them in developing decisive awareness and personality, maintaining emotional balance.

Learning Outcomes:

The students should be able to:

- Participate in Professional discussions by providing factual information, possible solutions, and examples.

Competencies:

- Comprehending key points of a topic and identifying main points including supporting details.
- Construct a logical chain of arguments and decisive points.
- Writing a review about a product by providing reasons, causes and effects

Sessions:

Based on Case Studies

1. Compare and Contrast
2. Cause and Effect
3. Problem and Solution

Unit-V: Drawing Conclusions

Unit Overview:

This module is intended to provide necessary inputs that enable the students to draw conclusions out of a discussion and provide reports.

Learning Outcomes:

Students should be able to:

- Provide logical conclusions to the topics under discussion.
- Prepare, present, and analyze reports.

Competencies:

- Reasoning skills - Coherent and logical thinking
- Reporting and Analyzing skills.
- Analyzing the points discussed.
- Connecting all points without gaps.
- Connectives
- Communicating the decisions

Sessions:

1. Report Writing
2. Reasoning
3. Analyzing
4. Generalization and Prediction
5. Précis writing

Reference Books:

1. Learn Correct English – A Book of Grammar, Usage and Composition by Shiv K. Kumar and Hemalatha Nagarajan, Pearson 2007.
2. Professional Communication by Aruna Koneru, McGraw Hill Education (India) Pvt. Ltd, 2016.
3. Technical Communication by Meenakshi Raman & Sangeeta Sharma, Oxford University Press 2009.
4. Technical Communication by Paul V. Anderson. 2007. Cengage Learning Pvt. Ltd., New Delhi.
5. English Vocabulary in Use series, Cambridge University Press 2008.
6. Handbook for Technical Communication by David A. Mc Murrey & amp; Joanne Buckley. 2012, Cengage Learning.
7. Communication Skills by Leena Sen, PHI Learning Pvt Ltd., New Delhi, 2009.
8. Job Hunting by Colm Downes, Cambridge University Press 2008.
9. English for Technical Communication for Engineering Students, Aysha Vishwamohan, Tata Mc Graw-Hill 2009.
10. Effective Technical Communication by M Asharaf Rizvi. McGraw Hill Education (India) Pvt. Ltd. 2nd Edition.

L	T	P	C
0	0	2	0

III Semester Syllabus
MC351HS: Gender Sensitization Lab

(An Activity-based Course)

[Common to CSE, IT, CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

This course aims:

- To develop students' sensibility with regard to issues of gender in contemporary India.
- To provide a critical perspective on the socialization of men and women.
- To introduce students to information about some key biological aspects of genders.
- To expose the students to debates on the politics and economics of work.
- To help students reflect critically on gender violence.
- To expose students to more egalitarian interactions between men and women.

Course Outcomes

- Students will have developed a better understanding of important issues related to gender in contemporary India.
- Students will be sensitized to basic dimensions of the biological, sociological, psychological and legal aspects of gender. This will be achieved through discussion of materials derived from research, facts, everyday life, literature and film.
- Students will attain a finer grasp of how gender discrimination works in our society and how to counter it.
- Students will acquire insight into the gendered division of labour and its relation to politics and economics.
- Men and women students and professionals will be better equipped to work and live together as equals.
- Students will develop a sense of appreciation of women in all walks of life.
- Through providing accounts of studies and movements as well as the new laws that provide protection and relief to women, the textbook will empower students to understand and respond to gender violence.

Course Description

This course offers an introduction to Gender Studies, an interdisciplinary field that asks critical questions about the meanings of sex and gender in society. The primary goal of this course is to familiarize students with key issues, questions and debates in Gender Studies, both historical and contemporary. It draws on multiple disciplines – such as literature, history, economics, psychology, sociology, philosophy, political science, anthropology and media studies – to examine cultural assumptions about sex, gender, and sexuality. This course integrates analysis of current events through student presentations, aiming to increase awareness of contemporary and historical experiences of women, and of the multiple ways that sex and gender interact with race, class, caste, nationality and other social identities. This course also seeks to build an understanding and initiate and strengthen programmes combating gender-based violence and discrimination. The course also features several exercises and reflective activities designed to examine the concepts of gender, gender-based violence, sexuality, and rights. It will further explore the impact of gender-based violence on education, health and development.

Unit-I: Understanding Gender

Introduction: Definition of Gender-Basic Gender Concepts and Terminology-Exploring Attitudes towards Gender-Construction of Gender-Socialization: Making Women, Making Men - Preparing for Womanhood, Growing up Male.

Unit-II: Gender Roles and Relations

Two or Many? -Struggles with Discrimination-Gender Roles and Relations-Types of Gender Roles-Gender Roles and Relationships Matrix-Missing Women-Sex Selection and Its Consequences-Declining Sex Ratio. Demographic Consequences-Gender Spectrum: Beyond the Binary.

Unit-III: Gender and Labour

Division and Valuation of Labour-Housework: The Invisible Labor - “My Mother doesn’t Work.” “Share the Load.”-Work: Its Politics and Economics -Fact and Fiction. Unrecognized and Unaccounted work - Gender Development Issues-Gender, Governance and Sustainable Development-Gender and Human Rights-Gender and Mainstreaming.

Unit–IV: Gender - Based Violence

The Concept of Violence- Types of Gender-based Violence-Gender-based Violence from a Human Rights Perspective-Sexual Harassment: Say No! -Sexual Harassment, not Eve-teasing-Coping with Everyday Harassment- Further Reading: “Chupulu”.

Domestic Violence: Speaking Out -Is Home a Safe Place? -When Women Unite [Film]. Rebuilding Lives. Thinking about Sexual Violence Blaming the Victim-“I Fought for my Life….”.

Unit – V: Gender and Culture

Gender and Film-Gender and Electronic Media-Gender and Advertisement-Gender and Popular Literature- Gender Development Issues-Gender Issues-Gender Sensitive Language-Gender and Popular Literature - Just Relationships: Being Together as Equals Mary Kom and Onler. Love and Acid just do not Mix. Love Letters. Mothers and Fathers. Rosa Parks-The Brave Heart.

Note: Since it is Interdisciplinary Course, Resource Persons can be drawn from the fields of **English Literature** or Sociology or Political Science or **any other qualified faculty who has expertise in this field from engineering departments.**

- Classes will consist of a combination of activities: dialogue-based lectures, discussions, collaborative learning activities, group work and in-class assignments. Apart from the above prescribed book, Teachers can make use of any authentic materials related to the topics given in the syllabus on "Gender".

Suggested Readings:

- The Textbook, "Towards a World of Equals: A Bilingual Text Book on Gender" written by A. Suneetha, Uma Bhrugubanda, DuggiralaVasanta, Rama Melkote, Vasudha Nagaraj, Asma Rasheed, Gogu Shyamala, Deepa Sreenivas and Susie Tharu published by Telugu Akademi, Telangana Government in 2015.

Assessment and Grading:

- Discussion & Classroom Participation: 20%.
- Project/Assignment: 30%.
- End Term Exam: 50%.

IV Semester

S.No.	Course Code	Course Title	Instructions			Examination			Credits
			Hours per work			Max. Marks		Duration of SEE in Hours	
			L	T	P/D	CIE	SEE		
1	MS401HS	Business Economics and Financial Analysis	3	0	0	30	70	3	3
2	CS402PC	Principles of Operating Systems	3	0	0	30	70	3	3
3	CS404PC	Introduction to Database Management Systems	3	1	0	30	70	3	4
4	CS406PC	Object Oriented Programming using Java	3	1	0	30	70	3	4
5	CS410PC	Formal Language and Automata Theory	3	0	0	30	70	3	3
6	MC401HS	Constitution of India	3	0	0	30	70	3	0
7	CS452PC	Principles of Operating Systems Lab	0	0	2	30	70	3	1
8	CS454PC	Introduction to Database Management Systems Lab	0	0	2	30	70	3	1
9	CS455PC	Java Programming Lab	0	0	2	30	70	3	1
10	EN452HS	Finishing School-II	0	0	2	30	70	3	1
Total Hours/Marks/Credits			**18**	**2**	**8**	**300**	**700**	**--**	**21**

L: Lecture **T**: Tutorial **D**: Drawing **P**: Practical

CIE - Continuous Internal Evaluation **SEE -** Semester End Examination

L	T	P	C
3	0	0	3

IV Semester Syllabus

MS401HS: Business Economics and Financial Analysis

[Common to CSE, IT, CSE (AI&ML) & CSE (DS)]

Course Objectives

- Students will understand various forms of Business and the impact of economic variables on the business, concepts of Business economics and its significance.
- Gain the knowledge on various market dynamics namely Demand, elasticity of demand, and demand forecasting.
- To disseminate the knowledge on production function, laws of production, Market structures, while dealing with the concept of cost and breakeven analysis.
- To acquaint the students regarding Accounting and various books of accounts.
- To enable the students to analyze a company's financial statements through ratios and come to a reasoned conclusion about the financial situation of the company.

Course Outcomes

- Students will have ability to start a suitable business organization with available resources.
- Analyze various aspects of Demand, Elasticity of demand and Demand Forecasting.
- Will be associated with different market structures, production theories, and cost variables and pricing objectives and methods.
- Will learn preparation of Financial Statements.
- Will analyze financial well-being of the business while using ratios.

Unit – I: Introduction to Business and Economics

Economics: Significance of Economics, Micro and Macro Economic Concepts, National Income - Concepts and Importance, Inflation, Business Cycle - Features and Phases.

Business: Structure of Business Firm, Types of Business Entities – Sole Proprietorship – Partnership – Cooperative Societies - Limited Liability Companies, Sources of Capital – Conventional sources and Non-Conventional Sources of Finance.

Business Economics: Nature and Scope of Business Economics, Role of Business Economist, Multidisciplinary nature of Business Economics.

Unit – II: Demand and Supply Analysis

Demand Analysis: Demand - Meaning, Determinants of Demand, Law of Demand, Exceptions of Law of Demand, Demand Function, Changes in Demand – Increase and decrease in Demand - Extension and Contraction in Demand

Elasticity of Demand: Elasticity – Meaning, Types of Elasticity – Price Elasticity – Income Elasticity – Cross Elasticity–Advertising Elasticity of Demand, Factors affecting Elasticity of

Demand, Measurement and Significance of Elasticity of Demand, Elasticity of Demand in decision making

Demand Forecasting: Characteristics of Good Demand Forecasting, Steps in Demand Forecasting, Methods of Demand Forecasting – Survey methods, Statistical methods

Supply Analysis: Supply – Meaning, Determinants of Supply, Supply Function & Law of Supply.

Unit- III: Production, Cost, Market Structures & Pricing

Production Analysis: Production – Meaning, Factors of Production, Production Function, Production Function with one variable input, two variable inputs, Returns to Scale, Cobb-Douglas production function

Cost analysis: Cost–Meaning, Types of Costs, Short run and Long run Cost Functions.

Market Structures: Nature of Competition, Features of Perfect competition, Monopoly, Oligopoly, Monopolistic Competition.

Pricing: Pricing -Meaning, Objectives of pricing, pricing methods – Cost based pricing methods – Demand based pricing methods – Competition based pricing methods – Strategy based pricing methods - Product Life Cycle based Pricing, Break Even Analysis (simple problems), Cost Volume Profit Analysis.

Unit-IV: Financial Accounting

Accounting concepts and Conventions, Accounting Equation, Double-Entry system of Accounting, Rules for maintaining Books of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, Preparation of Final Accounts along with adjustments– Trading account – Profit and loss account – Balance sheet (simple problems)

Unit – V: Financial Analysis through Ratios

Concept of Ratio Analysis, Importance, Liquidity Ratios- Current Ratio – Quick Ratio – Absolute Liquid Ratio, Profitability Ratios – Gross Profit Ratio – Net Profit Ratio – Operating Ratio, Turnover Ratios – Stock Turnover Ratio – Debtors Turnover Ratio – Creditors Turnover Ratio, Leverage Ratios – Debt-to-Assets Ratio - Debt-Equity Ratio - Proprietary Ratios and interpretation (simple problems).

Suggested Readings:

1. D.D. Chaturvedi, S.L. Gupta, Business Economics - Theory and Applications, International Book House Pvt. Ltd. 2013.
2. Dhanesh K Khatri, Financial Accounting, Tata McGraw Hill, 2011.
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2nd edition, Tata McGraw Hill Education Pvt. Ltd. 2012.

Reference Books:

1. P. L. Mehta, Managerial Economics, Analysis, Problems & Cases, 8th Edition, 2001, Sultan Chand & Sons.
2. S.N. Maheshwari, Sunil K Maheshwari, Sharad K Maheshwari, Financial Accounting, 5e, Vikas Publications, 2013.

L	T	P	C
3	0	0	3

IV Semester Syllabus
CS402PC: Principles of Operating Systems

[Common to CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- To understand the OS role in the overall computer system and study different OS and compare their features.
- To understand the scheduling policies of OS and introduces system call Interface for process management.
- To understand process concurrency and synchronization
- To understand the different memory management techniques
- To understand the concepts of input/output, file management and Introduces system call for file management

Course Outcomes

- Ability to work in different Computing Environments.
- Learn about minimization of turnaround time, waiting time and response time and also maximization of throughput by keeping CPU as busy as possible.
- Ability to design and solve synchronization problems.
- Acquires the knowledge to optimize the memory usage.
- Ability to change Access control to protect files.

Unit – I

Operating System – Introduction: Operating system objectives, User view, System view, Operating system definition, Computer System Architecture, OS Operations, Process Management, Memory Management, Storage Management, Protection and Security, Computing Environments.

System Structures: Operating System services, User and OS Interface, System Calls, Types of System Calls, System Programs, Operating System Design and Implementation, OS Structure.

Unit – II

Process and CPU Scheduling –Process concept**:** The Process, Process State, PCB, Threads. Process Scheduling-Scheduling Queues, Schedulers, Context Switch, and Operations on Processes.

Interprocess Communication Mechanisms: IPC between processes on a single computer system, IPC between processes on different systems using pipes, FIFOs, message queues, shared memory.

System call interface for process management - fork(), exit(), wait(), waitpid(), exec().

Unit – III

Process Management and Synchronization - The Critical Section Problem, Peterson's solution, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Monitors.

Deadlocks - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock

Unit – IV

Memory Management and Virtual Memory: – Memory Management Strategies-Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of Page Table.

Virtual Memory Management-Background, Demand Paging, Copy-on-Write, Page Replacement, Page Replacement Algorithms-FIFO, Optimal, LRU, Allocation of Frames, Thrashing.

Unit – V

File System Interface and Operations -Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management.

System Calls-Usage of open(), create(), read(), write(), close(), lseek(), stat(), ioctl() system calls.

Study of Different Operating Systems: Windows, Unix and Android.

Suggested Readings:

1. Abraham Silberchatz, Peter B. Galvin ,Operating System Principles-, Greg Gagne 9th Edition, John Wiley
2. Operating Systems – Internals and Design Principles, W. Stallings, 7th Edition, Pearson.

Reference Books:

1. Crowley, TMH, Operating System A Design Approach.
2. Andrew S, Modern Operating Systems, Tanenbaum 2nd edition, Pearson/PHI.
3. W.R. Stevens, Advanced programming in the UNIX environment, Pearson education.

L	T	P	C
3	1	0	4

IV Semester Syllabus

CS404PC: Introduction to Database Management Systems

[Common to CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- To understand the relational database design principles.
- To apply the concurrency control, recovery, and indexing for the real time data
- To become familiar with database storage structures and access techniques

Course Outcomes

- Design a database using ER modelling.
- Develop complex queries using SQL.
- Apply normalization techniques on databases.
- Explain the ACID properties of transactions and apply the serializability tests.
- Solve problems using various indexing and hashing techniques.

Unit- I

Introduction: Purpose of Database Systems, View of Data, Database Languages, Database Models, Database Architecture, Database System Applications.

Introduction to Database design: Database Design and ER diagrams, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model.

Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data.

Unit-II

Relational Algebra and Calculus: Relational Algebra, Relational calculus – Tuple relational Calculus, Domain relational calculus.

SQL: Queries, Constraints, Form of Basic SQL Query, UNION,INTERSECT, and EXCEPT, Nested Queries, Aggregate Operators, NULL values Complex Integrity Constraints in SQL, Introduction to Views, Destroying /Altering Tables and Views, Triggers.

Unit-III

Functional Dependencies -Introduction, Basic Definitions, Trivial, Non Trivial functional dependencies, Closure of set of dependencies, Closure of Attributes

Schema Refinement: Problems caused by redundancy, decompositions, Properties of decomposition, Normalization- FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multi-valued dependencies, FOURTH normal form, FIFTH normal form.

Unit-IV

Transaction Management and Recovery: Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log–Based Recovery

Unit-V

Storage and Indexing: Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree based Indexing, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure-Insertion , Deletion and Searching.

Suggested Readings:

1. Raghu Ramakrishnan, Johannes Gehrke, Data base Management Systems, McGraw Hill Education (India) Private Limited, 3rd Edition.
2. A. Silberschatz, Henry. F. Korth, S. Sudarshan, Data base System Concepts, McGraw Hill Education(India) Private Limited, 6th edition.

Reference Books:

1. R Elmasri, Shamkant B.Navathe, Database Systems, 6th edition, Pearson Education.
2. Peter Rob & Carlos Coronel, Database System Concepts, Cengage Learning.
3. M. L. Gillenson and others, Introduction to Database Management, Wiley Student Edition.
4. Lee Chao, Auerbach publications, Taylor & Francis Group,
5. C. J. Date, Database Development and Management, Introduction to Database Systems, Pearson Education.

L	T	P	C
3	1	0	4

IV Semester Syllabus
CS406PC: Object Oriented Programming using Java

[Common to CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- To introduce and understand Object-Oriented Programming concepts, and apply them in solving problems.
- To introduce the principles of Inheritance and Polymorphism; and demonstrate how they relate to the design of Abstract classes.
- To introduce the implementation of Packages and Interfaces.
- To introduce the concepts of Exception handling and Multithreading.
- To introduce the design of Graphical User Interface using Applets and Swing controls.

Course Outcomes

- Able to model and solve real world problems using OOP techniques.
- Able to understand the use of Abstract classes.
- Able to solve problems using java Collection framework and I/O classes.
- Able to develop Multi-threaded applications with synchronization.
- Able to develop desktop application with rich graphical interface.

Unit – I

Object-Oriented Thinking- Introduction to Object-Oriented concepts, Java buzzwords, An Overview of Java, Data types, Variables and Arrays, operators, expressions, control statements, Introducing classes: Class fundamentals, Declaring objects, introducing Methods, Constructors, and this keyword, method overloading, String handling.
Inheritance– Inheritance basics, using super, method overriding, dynamic method dispatch, using abstract class, using final with inheritance and Object class.

Unit – II

Packages- Defining a Package, CLASSPATH, Access protection, importing packages.
Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces, extending interfaces.
Exception Handling - Fundamentals of exception handling, Exception types, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.

Unit – III

Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, synchronizing threads, inter thread communication.

Stream based I/O (java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, The Console class, Serialization, Enumerations, autoboxing, generics.

Unit – IV

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- ArrayList, LinkedList, HashSet, TreeSet, Accessing a Collection via an Iterator, Using an Iterator, The for-each alternative, Map interfaces and classes, Comparators, Collection Algorithms, Arrays, The Legacy Classes: Stack, Vector, HashTable, Dictionary, Properties

More Utility classes: StringTokenizer, Random, Scanner.

Unit – V

Applets – Applets and Applications, Simple applet, passing parameters to applets, Security Issues.

GUI Programming –java.awt – Components, Containers, limitations of AWT, MVC Architecture.

Swing Applications- Creating a Swing Applet, Painting in Swing- an example

Event Handling- The Delegation event model- Event, Event sources, Event Listeners, Event classes, Handling Mouse and Keyboard events, Adapter classes, Inner classes, Anonymous Inner classes.

Understanding Layout Managers- FlowLayout, BorderLayout, GridLayout, CardLayout, GridBagLayout.

Suggested Readings:

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

Reference Books:

1. J. Nino and F.A. Hosch, An Introduction to programming and OO design using Java, John Wiley & Sons.
2. Y. Daniel Liang, Introduction to Java programming, Pearson Education.
3. P. Radha Krishna Object Oriented Programming through Java, University Press.
4. S. Malhotra, S. Chudhary, Programming in Java, 2nd edition, Oxford Univ. Press.
5. R. A. Johnson, Cengage Learning Java Programming and Object-oriented Application Development.

L	T	P	C
3	0	0	3

IV Semester Syllabus
CS410PC: Formal Languages and Automata Theory
[Common to CSE (AI&ML) & CSE (DS)]

Course Objectives

- To provide introduction to some of the central ideas of theoretical computer science from the perspective of formal languages.
- To introduce the fundamental concepts of formal languages, grammars and automata theory.
- To classify machines by their power to recognize languages.
- To employ finite state machines to solve problems in computing.
- To understand deterministic and non-deterministic machines.
- To understand the differences between decidability and undecidability.

Course Outcomes

- Able to understand the concept of abstract machines and their power to recognize the languages.
- Able to employ finite state machines for modeling and solving computing problems.
- Able to design context free grammars for formal languages.
- Able to distinguish between decidability and undecidability.
- Able to gain proficiency with mathematical tools and formal methods.

Unit – I

Introduction to Finite Automata: Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems.

Deterministic Finite Automata: Definition of DFA, How a DFA Process Strings, The language of DFA.

Nondeterministic Finite Automata: Formal Definition, an application: Text Search, Conversion of NFA to DFA.

Finite Automata with Epsilon-Transitions: Formal Definition, epsilon- closure, conversion of NFA with Є-transitions to DFA.

Moore and Mealy machines: Definition of Moore and Mealy machines, conversion of Mealy to Moore machine and vice-versa.

Unit – II

Regular Expressions: Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions.

Pumping Lemma for Regular Languages: Statement of the pumping lemma, Applications of the Pumping Lemma.

Closure Properties of Regular Languages: Closure properties of Regular languages, Decision Properties of Regular Languages, Equivalence and Minimization of Automata.

Unit – III

Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Sentential Forms, Parse Tree, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages.
Push Down Automata: Definition of the Pushdown Automaton, the Language of a PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automata.

Unit – IV

Normal Forms for Context- Free Grammars: Eliminating useless symbols, Eliminating Unit productions, Eliminating Є-productions, Chomsky Normal form and Greibach Normal form.
Pumping Lemma for Context-Free Languages: Statement of pumping lemma, Applications.
Closure Properties of Context-Free Languages: Closure properties of CFL’s, Decision Properties of CFL's.

Unit – V

Turing Machines: Introduction to Turing Machine, Types of Turing machines, Formal Description, Instantaneous description, the language of a Turing machine, halting problem.
Undecidability: Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines, Recursive languages, Properties of recursive languages, Post's Correspondence Problem, Modified Post Correspondence problem, Other Undecidable Problems, Counter machines, Universal Turing machine.

Suggested Readings:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, 3rd Edition, Pearson Education.
2. K.L.P Mishra and N. Chandrashekaran, Theory of Computer Science – Automata, Languages and computation, 2nd Edition, PHI.

Reference Books:

1. John C Martin, Introduction to Languages and The Theory of Computation, TMH.
   Daniel I.A. Cohen, John Wiley, Introduction to Computer Theory,
2. P. K. Srimani, Nasir S. F. B, A Text book on Automata Theory, Cambridge University Press.
3. Michael Sipser, Introduction to the Theory of Computation, 3rd Edition, Cengage Learning.
4. Kamala Krithivasan, Rama R, Introduction to Formal languages Automata Theory and Computation, Pearson.

L	T	P	C
3	0	0	0

IV Semester Syllabus
MC401HS: Constitution of India

[Common to CSE, IT, CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- Students will get to know about the history of Indian Constitution
- Students will get to know about President election and his Powers
- Students will get to know about Council of Ministers and their election Procedure and their Powers and Responsibilities
- Students will get know about Judicial System in India
- Students will get know about Panchayat-raj System in India

Course Outcomes

- This enables the Students to know about the Rights of Citizen.
- This enables the Students to know about Fundamental Duties of People.
- This enables the Students to Know the Directive principles of State Policy.
- This enables the Students to know about Functioning of Parliament and its Powers.
- This enables the Students to know about various Constitutional bodies in India.

Course Content

1. Meaning of the constitution, law and constitutionalism
2. Historical perspective of the Constitution of India
   - Drafting Committee
3. Salient features and characteristics of the Constitution of India
   - Preamble
   - Salient Features
   - Major Sources of Indian Constitution
4. Scheme of the fundamental rights
   - Article 13 to 32
   - Scheme of the Fundamental Right to Equality
   - Scheme of the Fundamental Right to certain Freedom
   - Scope of the Right to Life and Personal Liberty
5. The scheme of the Fundamental Duties and its legal status
   - List of Fundamental Duties
   - Justifiability of Fundamental Duties
6. The Directive Principles of State Policy – Its importance and implementation
   - Categories - Gandhian, Socialist and Liberal Principles
   - Significance of Directive Principles of State Policy
   - Relation between Fundamental rights and Directive Principles of State Policy

7. Federal structure and distribution of legislative and financial powers between the Union and the States
   - Union List
   - State List
   - Concurrent List
   - Residuary Powers
8. Parliamentary Form of Government in India.
9. The constitutional powers and status of the President of India vs the constitutional powers and status of the Council of ministers headed by the Prime Minister
10. Amendment of the Constitution and its Procedure
    - Procedure of Amendment to Constitution of India
    - Important Amendments
11. Emergency Provisions: National Emergency, President Rule, Financial Emergency
12. Local Self Government – Constitutional Scheme in India
    - Urban local Self Government
    - Rural local Self Government
13. Important Constitutional Bodies
    - Election Commission of India
    - Finance Commission of India
    - Union Public Service Commission
    - C-AG

Suggested Readings:

1. Subhash Kashyap, Our Constitution, National Book Trust, 5th Edition, Reprint- 2017.
2. V. N Shukla, The Constitution of India, Law literature Publication, 11th Edition, 2020.

Reference Books:

1. M P Jain, Indian Constitutional Law, Lexis Nexis, 8th Edition, 2018.
2. Samaraditya Pal, Indian Constitution-Origin& Evolution, Lexis Nexis, 1st Edition, 2019.

L	T	P	C
0	0	2	1

IV Semester Syllabus
CS452PC: Principles of Operating Systems Lab
[Common to CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- To provide an understanding of the design aspects of operating system concepts through simulation
- Introduce basic Unix commands, system call interface for process management,
- Inter process communication and I/O in Unix

Course Outcomes

- Simulate and implement operating system concepts such as scheduling, deadlock.
- Simulate and implement process management, file management and memory management.
- Able to implement C programs using Unix system calls.

List of Experiments:

1. Hands on UNIX/LINUX Commands.
2. Shell programming for file handling.
3. Shell Script programming using the commands grep, awk, and sed.
4. Study pthreads and implement the following: Write a program which shows the performance improvement in using threads as compared with process.(Examples like Matrix Multiplication, quick sort, Merge sort)
5. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, fcntl, seek, stat, opendir, readdir)
6. Write programs using the Process system calls of UNIX/LINUX operating system (fork, exit, wait, waitpid, exec)
7. Implement the following CPU Scheduling algorithms with Pthreads
   a) FCFS b) SJF c) Round Robin d) priority
8. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
9. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
10. Write C programs to simulate the following memory management techniques with Pthread library
    a) Paging b) Segmentation
11. Write C programs to illustrate the following IPC mechanisms
    a) Pipes b) FIFOs c) Message Queues d) Shared Memory

Reference Books:

1. William Stallings, Operating Systems – Internals and Design Principles, Fifth Edition–2005, Pearson Education/PHI
2. Crowley, Operating System - A Design Approach- TMH.
3. Andrew S Tanenbaum, Modern Operating Systems, 2nd edition, Pearson/PHI
4. Kernighan and Pike, UNIX Programming Environment, PHI/Pearson Education
5. U. Vahalia, UNIX Internals: The New Frontiers, Pearson Education.

L	T	P	C
0	0	2	1

IV Semester Syllabus

CS454PC: Introduction to Database Management Systems Lab

[Common to CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- Introduce ER data model, database design and normalization
- Learn SQL basics for data definition and data manipulation

Course Outcomes

- Apply the built-in functions and write simple queries on various databases.
- Create, insert and manipulate data using SQL commands.
- Develop complex queries using joins & nested queries.
- Add constraints on databases and implement DCL, TCL and advanced SQL commands.
- Develop programs using cursors, triggers, exceptions, procedures and functions in PL/SQL.

List of Experiments

1. Database Design with E-R Model
2. Relational Model (Practicing DDL,DML, DCL commands)
3. Applying constraints on Relations
4. Queries using - Aggregate functions, Order By, GROUP BY, HAVING Clause
5. Number Functions, String Functions, Date Functions
6. Nested Queries, Querying (using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT etc.)
7. Creation and dropping of Views.
8. Working with Join operations
9. Triggers (Creation of insert trigger, delete trigger, update trigger)
10. Procedures and Functions, Packages, Exceptions
11. Usage of Cursors

Reference Books:

1. S.Shah and V. Shah, The X Team, Oracle for Professionals, SPD
2. Shah, Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, PHI.
3. M. L. Gillenson, Fundamentals of Database Management Systems, Wiley Student Edition.

L	T	P	C
0	0	2	1

IV Semester Syllabus
CS455PC: Java Programming Lab
[Common to CSE, IT, CSBS, CSE (AI&ML) & CSE (DS)]

Course Objectives

- To write programs using abstract classes.
- To write multithreaded programs.
- To write programs for solving real world problems using java collection framework.
- To write GUI programs using swing controls in Java.
- To introduce java compiler and eclipse platform.
- To impart hands on experience with java programming.

Course Outcomes

- Able to write programs using abstract classes.
- Able to write multithreaded programs.
- Able to write GUI programs using swing controls in Java.
- Able to write programs for solving real world problems using java collection frame work.

Note:

1. Use LINUX and MySQL for the Lab Experiments. Though not mandatory, encourage the use of Eclipse platform.
2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

List of Experiments:

1. Use Eclipse or Net bean platform and acquaint with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.
2. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order.
3. Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set of integers.
4. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method printArea () that prints the area of the given shape.
5. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread

computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

6. Write a Java program that correctly implements the producer – consumer problem using the concept of interthread communication.
7. Write a Java program for the following: Create a doubly linked list of elements. Delete a given element from the above list. Display the contents of the list after deletion.
8. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).
9. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException. Display the exception in a message dialog box.
10. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.
11. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in selected color. Initially, there is no message shown.
12. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.
13. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).
14. Develop an applet in Java that displays a simple message.
15. Develop an applet in Java that receives an integer in one text field, and computes its factorial value and returns it in another text field, when the button named “Compute” is clicked.

Reference Books:

1. P. J. Deitel and H. M. Deitel, Java for Programmers, 10th Edition Pearson education.
2. Bruce Eckel, Thinking in Java, Pearson Education.
3. D. S. Malik and P. S. Nair, Java Programming, Cengage Learning.
4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

L	T	P	C
0	0	2	1

IV Semester Syllabus
EN452HS: Finishing School- II

(Common to all Branches)

Course Overview

In view of the growing importance of English as a tool for global Communication and the consequent emphasis on training students to acquire language skills, this syllabus has been designed to develop linguistic, communicative and critical thinking competencies of Engineering students.

Course Objectives

The main objective of this finishing school curriculum is to provide content for developing the LSRW skills of language learning and to facilitate proficiency in both receptive and productive skills, among students.

Methodology:

- Students will be given Reading/Listening exercises that they would have to do as a prerequisite for the class room intervention
- Every Session will have activities on all the four skills. Listening, Speaking, Reading and Writing
- Vocabulary exercises will also be part of every session
- Students will be asked to summarize their takeaways in every class in three sentences.
- The students will be given a self study plan for language enhancement and will be given extra reading and writing exercises as and when necessary.
- To personalize learning, a variety of case studies and structured problem solving activities will be given in small groups and the trainers will facilitate peer reviews.

Unit-I: Discussions and Debates

Module Overview:

The module enables the students to build strategies for effective group interaction. It focuses on developing decisive awareness and positive personality while maintaining emotional balance.

Learning Outcomes:

The students should be able to:

- Participate in group discussions by providing factual information, real time solutions, and examples.
- Debate on a topic by picking up the key points from the arguments offered.

Competencies:

- Analytical and Probing Skills
- Interpersonal Skills

- Identifying key points of the debate.
- Problem solving ability
- Constructing a logical chain of arguments and presenting winning view points.

Sessions:

1. Six Thinking Hats
2. Initiation Techniques
3. Generating points
4. Summarization Techniques

Unit-II: Powerful Presentations

Unit Overview:

Presentations need to be clear and logical. This Module is designed to introduce students to an ideal structure for a presentation

Learning Outcomes:

Students should be able to:

- Prepare, present, and analyze reports
- Analyze the points discussed
- Connect all points logically with coherence
- Connectives
- Communicate the decisions
- Provide logical conclusions

Sessions:

1. Persuasion skills
2. Cultivate appropriate body language and group dynamics
3. Debating Structure and Content
4. Case Study based Group Discussions

Unit-III: Effective Technical Writing

Unit Overview:

Organizing the writing in a logical order, using headings, linkers and sequence markers. This module is designed to give the students inputs on how to organize using Information Mapping. The students are also given inputs to correct spelling, language and Punctuation errors, as part of editing.

Learning Outcomes:

The Students should be able to choose appropriate words and tone to presentaccurate, specific, and factual written documents

Competencies:

- Reporting an incident
- Writing/Presenting an essay
- Language and Vocabulary

Sessions:

1. Information Mapping
2. Report writing
3. Memos

4. SoP (Statement of Purpose)
5. MoM (Minutes of the Meeting)

Unit-IV: Reading for Content and Context

Unit Overview:

This course is designed to develop and improve reading and study skills needed for employability. Topics include identifying main idea and supporting details, determining author's purpose and tone, distinguishing between fact and opinion, identifying patterns of organization in a paragraph or passage and the transition words associated with each pattern. Also recognizing the relationship between sentences, puzzling out meanings in context, identifying logical inferences and conclusions.

Learning Outcomes:

Upon completion of the course, students should be able to:

1. Compose a summary of a given text.
2. Apply reading skills appropriate to different genres

Competencies

- Distinguish facts from opinions.
- Make inferences
- Identify author's purpose, point of view, tone, and perspective.
- Comprehend the use of figurative language.
- Synthesize information gathered from reading in order to give informedopinion.

Sessions:

1. Skimming and Scanning Techniques
2. Recognition of author's purpose
3. Awareness of stylistic differences
4. Evaluation and Discernment of fact and opinion

Unit-V: Critical Reading Skills

Unit Overview:

Research shows that good reading skills can lead to well written assignments. In this unit, students will learn reading strategies to understand and retain information, organization of reading passages, and strategies for learning and retaining vocabulary. Building on these basic strategies, students will develop skills to critically analyze texts. In addition, students will practice and develop paraphrasing and summarizing skills. Students' feedback is integral to the learning process.

Learning Outcomes:

- Recognition of propaganda techniques
- Present vocabulary building methods
- Use comprehension and vocabulary strategies to improve reading skills.

Competencies:

The students will develop enhanced ability to apply the following critical thinking skills when reading:

a. Understand the meaning of new vocabulary through:

1. Context clues, e.g., synonyms, antonyms, examples, definitions, and restatements, etc.

2. Roots and affixes

b. Analyze text (simple outlining and note taking) summarize, draw conclusions, and apply information to personal experiences.

Sessions

1. Contextual Vocabulary-One-word substitutes
2. Homophones, Homonyms and Homographs
3. Idioms and Phrases
4. Synonyms, Antonyms and Phrasal verbs
5. Note making and Inference
6. Main idea identification
7. Précis Writing.

Reference Books:

1. Learn Correct English – A Book of Grammar, Usage and Composition by Shiv K. Kumar and Hemalatha Nagarajan, Pearson 2007.
2. Professional Communication by Aruna Koneru, McGraw Hill Education (India) Pvt. Ltd, 2016.
3. Technical Communication by Meenakshi Raman & Sangeeta Sharma, Oxford University Press 2009.
4. Technical Communication by Paul V. Anderson. 2007. Cengage Learning Pvt. Ltd. New Delhi.
5. English Vocabulary in Use series, Cambridge University Press 2008.
6. Handbook for Technical Communication by David A. Mc Murrey & Joanne Buckley, 2012 Cengage Learning.
7. Communication Skills by Leena Sen, PHI Learning Pvt Ltd., New Delhi, 2009.
8. Job Hunting by Colm Downes, Cambridge University Press 2008.
9. English for Technical Communication for Engineering Students, Aysha Vishwamohan, Tata Mc Graw-Hill 2009.
10. Effective Technical Communication by M Asharaf Rizvi. McGraw Hill Education (India) Pvt. Ltd., 2nd Edition.