LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY
(An Autonomous Institution)
DEPARTMENT OF INFORMATION TECHNOLOGY
SCHEME OF INSTRUCTIONS & EXAMINATIONS [LR-21]
(W.E.F Academic Year 2022-23)
B.E. III-Semester

S. No.	Course Code	Course Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/ D	Contact Hours/Week	CIE	SEE	Duration in Hours	
Theory Course											
1	U21EN301	HSMC	English For Technical Communication	2	-	-	2	40	60	3	2
2	U21EC305	ESC	Digital Logic Design	3	-	-	3	40	60	3	3
3	U21MA301	BSC	Mathematics-III (Probability and Statistics)	3	1	-	4	40	60	3	4
4	U21CS302	PCC	Data Structures	3	-	-	3	40	60	3	3
5	U21CM301	PCC	Python Programming	3	-	-	3	40	60	3	3
Practical/ Laboratory Course											
6	U21EN3L1	HSMC	Advanced Communication Skills Lab	-	-	3	3	25	50	3	1.5
7	U21EC3L4	ESC	MATLAB	-	-	3	3	25	50	3	1.5
8	U21CS3L1	PCC	Data Structures Lab	-	-	3	3	25	50	3	1.5
9	U21CM3L1	PCC	Python Programming Lab	-	-	3	3	25	50	3	1.5
Bridge Course*											
10	U21CS3L2	ESC	C Programming Lab	-	-	2	2	50	-	-	-
11	U21EN3L2	HSMC	Effective Communication Skills Lab	-	-	2	2	50	-	-	-
Total				**14**	**1**	**12 (*16)**	**27 (*31)**	**300 (*400)**	**500**	**-**	**21**

*** Bridge Course for Lateral Entry Admitted Students Only.**

L: Lecture*(Hrs/Wk/Sem)* **T**: Tutorial *(Hrs/Wk/Sem)* **P**: Practical / **D**: Drawing *(Hrs/Wk/Sem)*
CIE: Continuous Internal Evaluation **SEE**: Semester End Examination (Univ.Exam)
BSC: Basic Science Courses **ESC**: Engineering Science Courses **MA**: Mathematics
HSMC: Humanities and Social Sciences **PCC:** Programme Core Courses **EN:** English
IT: Information Technology **EC:** Electronics and Communication
CS: Computer Science **CM:** CSE-AIML

Note:
1. Each contact hour is a Clock Hour.
2. The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

Course Code	Course Title						Core/Elective
U21EN301	ENGLISH FOR TECHNICAL COMMUNICATION COMMON TO CSE(AIML),IT,MECH,AIML & ECE						Core
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
Basic English	L	T	D	P			
	2	-	-	-	40	60	2

Course Objectives

To expose the students to:

1. Understand the significance of Technical Writing.
2. Various aspects of professional communication.
3. Different types of business correspondence.
4. Various styles of technical report writing.
5. Designing, creating and developing technical manual.
6. Familiarize with the technical features of information transfer.

Course Outcomes

On successful completion of the course, the students would be able to:

1. Apply technical communication skills effectively
2. Adapt different types of official correspondence
3. Construct report writing using various techniques
4. Develop adequate skills of manual writing
5. Interpret the information transfer from verbal to non-verbal data and vice-versa

UNIT-I

Definition and Features of Technical communication: Definition, Types and Process of Communication, Definition and features of technical communication (precision, relevance, format, style, use of visual aids),Difference between general writing and technical writing, Types of technical communication.

UNIT-II

Technical Writing-I (Official correspondence): Emails, Business letters (all types), Business proposals, Preparation of Minute of Meeting.

UNIT-III

Technical writing-II (Reports): Definition, Importance, Types of Report - Memo, Letter & Manuscript, Feasibility report, Project report, Progress report, Evaluation report.

UNIT-IV

Technical writing-III (Manuals): Types of manuals, User manual, Product manual, Operation manual

UNIT-V

Information Transfer and Presentations: Non-verbal (bar diagram, flow chart, pie chart, tree diagram) to verbal (writing), Verbal (written) to non-verbal, Important aspects of oral and visual presentations.

Reference Books:

1. Raman, Meenakshi & Sharma, Sangeeta. (2015). Technical Communication: Principle sand Practice (3rd Ed.). New Delhi.
2. Rizvi, Ashraf, M. (2017).Effective Technical Communication (2nd Ed.).Tata McGraw Hill Education. New Delhi.
3. Sharma, R. C., & Mohan, Krishna. (2017). Business Correspondence and Report Writing: A Practical Approach to Business &Technical Communication (4th Ed.).Tata McGraw Hill Education. New Delhi.
4. Tyagi, Kavita & Misra, Padma. (2011). Advanced Technical Communication. New Delhi, PHIL earning.
5. English for Technical Communication for Engineering Students, AyshaVishwamohan, Tata McGraw-Hill 2009
6. Handbook for Technical Communication by David A. McMurrey& Joanne Buckley. 2012. Cengage Learning.

Course Code	CourseTitle						Core/Elective
U21EC305	DIGITAL LOGIC DESIGN						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Basic Electronics	3	0	0	0	40	60	3

Course Objectives:

The objectives of this course are

1. To understand the basic building blocks of digital hardware and various minimization techniques.
2. To analyze and design the Combinational and Sequential circuits.
3. To understand the concept of State machine.

Course Outcomes:

On Successful completion of the course, student will be able to

1. Demonstrate the number system conversions and simplify Boolean functions.
2. Analyze and simplify Boolean expressions using karnaugh-maps and tabulation method.
3. Analyze and Design various combinational circuits
4. Understanding of various Sequential circuits
5. Analyze and Design of counters and understanding FSM.

UNIT – I:

Number Systems: Number systems, Complements of Numbers, Codes- Weighted and Non-weighted codes and its Properties, Parity check code and Hamming code.

Boolean algebra: Basic Theorems and Properties, Switching Functions- Canonical and Standard Form, Algebraic Simplification, Digital Logic Gates, EX-OR gates, Universal Gates, Multilevel NAND/NOR realizations.

UNIT – II

Minimization of Switching Functions: Introduction, SOP and POS, Implementation of logic functions using K-Map: three variable, four variable & five variable, Quine- McCluskey Tabular method.

UNIT – III

Combination Circuits : Adders- half adder, full adder, Subtractors, Comparators, Multiplexers, De-multiplexers, Encoders, Decoders and Code converters: BCD to 7-segment converter, BCD to Excess-3 converter.

UNIT – IV

Sequential circuits: Basic Architectural Distinctions between Combinational and Sequential circuits, SR Latch, Flip Flops: SR, JK, JK Master Slave, D and T Type Flip Flops, Excitation Table of all Flip Flops, Timing and Triggering Consideration, Conversion from one type of Flip-Flop to another.

UNIT –V

Counters & Registers: Registers, classification of Shift registers. Synchronous counter and Asynchronous counters, Decade counter.

Finite State Machine-Capabilities of FSM- State table state graph-State equivalence and machine minimization, Simplification of completely Specified machines, State Reduction- Partition method.

Text Books:

1. Morris Mano M. and Michael D. Ciletti, "Digital Design, With an Introduction to Verilog HDL", Pearson 5 th edition, 2013.
2. ZVI Kohavi, "Switching and Finite Automata Theory", Tata McGraw Hill 2nd Edition, 1995.
3. RP Jain " Modern Digital Electronics" , Fourth Edition Mcgraw hill education (India) Pvt Limited, 2003

Suggested Reading:

1. Ronald J Tocci, Neal Widmer, Greg Moss, "Digital Systems: Principles and Applications", Pearson 11th Edition, 2011.
2. A Anand Kumar "Switching Theory and Logic Design" PHI 3rd Edition 2019.

Course Code	Course Title						Core/Elective
U21MA301	MATHEMATICS – III (PROBABILITY AND STATISTICS) COMMON TO CSE(AIML),IT, AIML , ECE & CSE						Core
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
Basic Mathematics	L	T	D	P			
	3	1	-	-	40	60	4

Course Objectives

The objective of the course is to:

1. Introduce the basic concepts of probability and statistics in engineering
2. Provide an overview of concepts of probability and statistics to engineers
3. Provide the knowledge of probability distributions , tests of significance
4. Acquire the concepts of curve fitting, correlation and regression.
5. Familiar with the concept of tests of hypothesis for decision making

Course Outcomes

After completing this course, the student will be able to:

1. Determine Probability, Random variables, distributions and its application
2. Apply the knowledge of some standard discrete probability distributions and moments
3. Calculate parameters of standard continuous probability distributions.
4. Find the parameters and concepts of correlation, regression and obtain the knowledge of sampling Theory with context to test of hypothesis.
5. Analyze and check the validity of statement using testing of hypothesis for various parameters and goodness of fit.

Unit-I

Introduction of Probability, Conditional probability, Baye's Theorem and its applications, Random variables, Types of random variables, Probability mass function and Probability density function, Mathematical expectations.

Unit-II

Discrete probability distributions: Binomial and Poisson distributions, Mean, variance, moment generating function and evaluation of statistical parameters for these distributions.

Unit-III

Continuous probability distributions, Uniform, Exponential and Normal distributions, Mean, variance and evaluation of statistical parameters for these distributions.

Unit-IV

Curve fitting by the method of least squares: fitting of straight lines, second degree parabolas and more general curves, Correlation, regression and Rank correlation. Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means.

Unit-V

Small Sample test for single mean, difference of means, test for ratio of variances, Chi-square test for goodness off it and independence of attributes, Low Rank Matrix, Singular Valued Decomposition (SVD).

Textbooks:

1. Advanced Engineering Mathematics, R.K.Jain&Iyengar, Narosa Publications.
2. B.S.Grewal,"Higher Engineering Mathematics",Khanna Publishers,2000.
3. Engineering Mathematics,P.Sivaramakrishna Das&C.VijayaKumar,Pearson India Education Services Pvt.Ltd.
4. Engineering Mathematics, SS Sastry, PHI Learning, Private Limited

REFERENCE BOOKS:

1. Fundamentals of Mathematical Statistics, S.C.Gupta&V.K.Kapoor, S.ChandPub.
2. P.G.Hoel, S.C.PortandC.J.Stone, "Introduction to Probability Theory", Universal Book Stall, 2003.
3. W.Feller,"An Introduction to Probability Theory anditsApplications",Vol.1, Wiley, 1968.
4. N.P.BaliandM.Goyal,"Atext book of Engineering Mathematics",Laxmi Publications,

Course Code	Course Title						Core/Elective
U21CS302	DATA STRUCTURES						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Programming for Problem Solving	3	-	-	-	40	60	3

Course Objectives:

Develop ability to

1. Develop skills to design and analyze simple linear and non linear datastructures,such as stacks, queues and lists and their applications.
2. Gainprogrammingskillstoimplementsortingandsearchingalgorithms
3. Strengthentheabilitytoidentifyandapplythesuitabledatastructuresforthe givenrealworld problem.
4. Gain knowledge in practical applications of data structures
5. Understand essential for future programming and software engineering courses.

Course Outcomes:

At the end of the course, student would be able to

1. Implementvariousdatastructuresusingarrays,linkedlists
2. DevelopADTnecessaryforsolvingproblemsbasedonStacksandQueues
3. Implementbinarytrees,generaltreestructures,advancedsearchtrees, heaps,graphs.
4. Implementhashfunctionsandhandlecollisions.
5. Implement various kinds of sorting techniques and apply appropriate techniques for solving a given problem

UNIT – I

Introduction Data Structures and Algorithms:

Introduction to data structures, classification of data structures, operations on data structures; Algorithm Specification, Recursive algorithms, Data Abstraction.

Performance analysis- Time Complexity and Space Complexity, Asymptotic Notation-Big O, Omega, and Theta notations.

UNIT – II

Stacks and Queues:

Stacks: Stack ADT, definition and operations, Implementations of stacks using array, applications of stacks, Arithmetic expression conversion and evaluation.

Queues: Queue ADT, definition and operations, Implementation of queues using Arrays, applications of linear queue, circular queue and double ended queue (deque).

UNIT – III

Linked Lists: Introduction, Singly linked list, representation of a linked list in memory, operations on a Singly Linked List, Implementation of Singly Linked List.

Doubly Linked Lists: Operations on Doubly Linked List, Implementation of Doubly Linked List , Circular linked list, Implementation of Stack and Queue using linked list.

UNIT – IV

Searching Techniques: Linear search and BinarySearchalgorithms.

Sorting Techniques: Bubble Sort, Insertionsort,Selection Sort, MergeSort, and QuickSort. Comparison among sorting techniques.

Hashing: StaticHashing,HashTables,HashFunctions,OverflowHandling,TheoreticalEvaluationof OverflowTechniques

UNIT – V

Trees: Introduction,BinaryTrees,TreeTraversals, ThreadedBinaryTrees, BinarySearch Tree, Heap Tree, AVL Tree.

Graphs:GraphAbstractDataType, Representation of Graph, GraphTraversals -DFSandBFS, Spanning Tree, Prim‘s and Kruskal‘sAlgorithms.

Suggested Readings:

1. “Fundamentals of Data Structures in C”, Ellis Horowitz, Sartaj Sahani, Susan Anderson Freed, Computer Science Press, 2004
2. S. Lipschutz, “Data Structures”, Tata McGraw Hill Education, 1st Edition, 2008.
3. D. Samanta, “Classic Data Structures”, PHI Learning, 2nd Edition, 2004.
4. Mark A Weiss, “Data Structures and Algorithm Analysis In C”, SecondEdition(2002),Pearson
5. “Data Structures and Algorithms in C++”, second Edition by Michael T. Goodrich and Roberto Tamassia.

Course Code	Course Title						Core/Elective
U21CM301	PYTHON PROGRAMMING						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Programming for Problem Solving	3	-	-	-	40	60	3

Course Objectives:

Develop ability to

1. Learn about Python programming language syntax, semantics, and the runtime environment
2. Familiarized with universal computer programming concepts like data types, containers
3. Acquire general computer programming concepts like conditional execution, loops & functions
4. Grasp the general coding techniques and object-oriented programming
5. Analyze about basic library modules.

Course Outcomes:

At the end of the course, student would be able to

1. Develop essential programming skills in computer programming concepts like data types, containers.
2. Apply the basics of programming in the Python language.
3. Solve coding tasks related conditional execution, loops.
4. Acquire coding tasks related to the fundamental notions and techniques used in object oriented programming
5. Write basic programs related to basic library modules.

UNIT- I

Introduction to Python: Data Types: Declaring and using Numeric data types: int, float, bool, complex, string data type and string operations, standard input and output functions, type conversion

Python Program Flow Control: Conditional blocks using if, else and elif, Control blocks using for loop, while loop. Loop manipulation using pass, continue, break and else.

UNIT- II

Functions: Creating, parameters and return values, Using Keyword Arguments and Default Parameter Values, Using Global Variables and Constants, recursive functions, lambda functions.

Python Data Structures: Lists- basic list operators, replacing, inserting, removing an element; searching and sorting lists, tuples.

UNIT-III

Dictionaries- literals, adding and removing keys, accessing and replacing values; traversing dictionaries. Sets and Strings operations.

Introduction to Object Oriented Programming: Class, object, attributes and methods; defining classes, inheritance, polymorphism, abstract classes, Exception handling.

UNIT- IV

Using Databases in Python:

Python MySQL Database Access, Create Database Connection, CREATE, INSERT, READ, UPDATE and DELETE Operation, DML and DDL Operation with Databases, Performing Transactions Handling, Database Errors.

UNIT- V

Python for Data Analysis:

Numpy: Introduction to numpy, Creating arrays, Using arrays and Scalars, Indexing Arrays, Array Transposition, Universal Array Function, Array Processing, Array Input and Output.

Pandas: What is pandas? Where it is used? Series in pandas, Index objects, Reindex , Drop Entry, Selecting Entries, Data Alignment, Rank and Sort Summary, Statics Missing Data, Index Hierarchy.

Suggested Readings:

1. Reema Thareja, "Python Programming using Problem Solving Approach", ISBN-13:978-0-19- 948017-3,Oxford University Press,2017
2. Vamsi kurama, "Python Programming: A modern approach", ISBN-978-93-325-8752- 6, Pearson,2018.
3. Mark Lutz , "Learning python", ISBN: 1-56592-464-9, Orielly, 4th edition, 1999.
4. W.Chun, "Core python programming", ISBN-13: 978-0132269933, Pearson, 2nd edition,2016.
5. Kenneth Lambert," Fundamentals of Python:First Programs",ISBN-13: 978- 1337560092,Cengage Learning publishers,First Edition,2012.
6. Allen B. Downey,"Think Python: How To Think Like A Computer Scientist", ISBN-13: 978- 1491939369, O'Reilly,2nd Edition,2016.
7. R Nageswara Rao, "Core Python Programming", Dreamtech press, 2017 Edition.
8. Mike Mc Grath "Python in easy steps: Makes Programming Fun", Kindle Edition, 2017.

Course code	Course Title						Core/Elective
U21EN3L1	ADVANCED COMMUNICATION SKILLS LAB COMMON TO CSE(AIML),IT,MECH,AIML & ECE						Core
Pre-requisites	Contact Hours Per Week				CIE	SEE	Credits
Basic English	L	T	D	P			
	-	-	-	3	25	50	1.5

Course Objectives:

To expose the students to:

1. Improve the students' fluency in English, through Interpersonal Communication skills.
2. Read the given text at normal speed and analyze and evaluate critically.
3. Exhibit their ability and skills relevantly and coherently through resume writing and cover letter writing.
4. Develop oral presentation skills to meet the global competition.
5. Boost confidence through the dynamics of Group Discussion.
6. Prepare all the students for their placements through Mock Interviews.

Course Outcomes:

On successful completion of the course the students would be able to:

1. Organize ideas relevantly and coherently in their communication
2. Analyze and Comprehend the text inferentially
3. Write Resume/CV and Cover letter effectively
4. Practice oral presentation confidently
5. Participate in group discussions dynamically and face interviews optimistically

List of Activities:

1. **Activities on Fundamentals of Inter-personal Communication:** Starting a conversation, responding appropriately and relevantly – using the right body language and Role Play in different situations.

2. Activities on Reading Comprehension: General vs. Local comprehension, reading for facts, guessing meanings from context, scanning, skimming, inferring meaning, critical reading.

3.Activities on Writing Skills: Structure and presentation of different types of Resume/CV writing, Cover letter writing, improving one's writing of Resume and Cover letter.

4. **Activities on Presentation Skills**: Oral presentations through JAM, Extempore, Seminars and Poster Presentations.

5. Activities on Group Discussion and Interview Skills: a).Dynamics of group discussion, intervention, summarizing, modulation of voice, body language, relevance, fluency and organization of ideas and rubrics for evaluation. b). Concept and Process, Pre-interview planning, opening strategies, answering strategies, interview (Types) and Mock Interviews.

Reference Books:

1. KoneruAruna .(2016). Professional Communication. Tata McGraw-Hill Publishing Company. Ltd, New Delhi
2. Raman, Meenakshi & Sharma, Sangeeta. (2015). Technical Communication: Principle sand Practice (3rd Ed.). New Delhi.
3. Anderson Paul V. (2007).Technical Communication. Wadsworth Cengage Learning Pvt. Ltd.
4. SenLeena. (2009). Communication Skills.PHI Learning Pvt Ltd., New Delhi,
5. DownesColm .(2008). Job Hunting.Cambridge University Press.

Course Code	Course Title						Core/Elective
U21EC3L4	MAT LAB						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Programming for Problem Solving Lab	-	-	-	3	25	50	1.5

Course Objectives:

Develop ability to:

1. Impart the knowledge with Mat lab software that enhances the programming skills for Research and Development
2. Create and troubleshoot Mat Lab Scripts.
3. Apply a variety of common numeric techniques to solve and visualize engineering related computational problems.
4. Perform basic operations in image processing applications.
5. Generate plots and export these for use in reports and presentation applications

Course Outcomes:

At the end of the course, the student would be able to :

1. Learn features of Mat lab as a programming language, its use as a simulation tool, and write simple programs to solve Scientific, Mathematics, and Engineering problems.
2. Generate Scripts and functions, and interactive computations in Mat lab development environment.
3. Perform and Compute different operations using Matlab.
4. Use basic flow control functions efficiently.
5. Create 2D and 3D plotting functions.

List of Experiments:

1. Introduction to MATLAB and creation of Mat lab file using basic commands.
2. Functional operations of Matrices.
3. Computing the mathematical expressions
4. Relational operations
5. Logical operations
6. Testing of Input/output Variables (Numbers and Strings)
7. Flow control functions
8. Complex and Statistical functions.
9. 2D Plotting
10. 3D Plotting

Suggested Reading:

1. Jaydeep Chakravarthy, 'Introduction to MATLAB Programming: Toolbox and Simulink',1/e, University Press,2014

Course Code	Course Title						Core/Elective
U21CS3L1	DATA STRUCTURES LAB						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Programming for Problem Solving Lab	-	-	-	3	25	50	1.5

Course Objectives:

Develop ability to

1. Understand essential concepts of simplelinearandnonlineardatastructures.
2. Analyze and implement programmingskillstoimplementsortingandsearchingalgorithms
3. Applythesuitabledatastructuresforthe givenrealworldproblems.
4. Acquireknowledgeinpracticalapplicationsofdatastructures.
5. Provide solutions for various graphical concepts.

Course Outcomes:

At the end of the course, student would be able to

1. Write programs in variousdatastructuresusingarrays andlinkedlists.
2. DevelopADTnecessaryforsolvingproblemsbasedonStacksandQueues
3. Evaluatebinarytrees,generaltreestructures,advancedsearchtrees, heaps,graphs.
4. Applyhashfunctionsandhandlecollisions
5. Implement various kinds of sorting techniques and apply appropriate techniques for solvingagiven Problem.

List of Experiments:

1. Implementationof Stacks and Queues using Arrays.
2. Solving Towers of Hanoi problem
3. Implementationof Circular Queue.
4. Solving tic-tac-toe problem
5. Implementationof InfixtoPostfixConversion,PostfixExpressionEvaluation.
6. ImplementationofSinglyLinkedList
7. Implementation of DoublyLinkedList.
8. Implementation of Circular LinkedList.
9. ImplementationofStacks,Queues using Linked Lists.
10. ImplementationofBinarySearchandHashing
11. Implementation of Operations on Binary Tree (Delete Entire Tree,Copy Entire Tree, Mirror Image, Level Order, Search for a Nodeetc.)
12. ImplementationofTree Traversals on Binary Trees.
13. ImplementationofBinarySearchTree. (Insertion, Deletion and Search operations)
14. ImplementationofoperationsonAVLTrees.
15. ImplementationofTraversalonGraphs.
16. ImplementationofSelection,Merge,QuickandInsertionSort.
17. Implementation of Prims and Kruskals Algorithm.

Suggested Readings:

1. S. Lipschutz, “Data Structures”, Tata McGraw Hill Education, 1st Edition, 2008.
2. D. Samanta, “Classic Data Structures”, PHI Learning, 2nd Edition, 2004.
3. Mark A Weiss, Data Structures and Algorithm Analysis In C, SecondEdition(2002),Pearson.

Course Code	Course Title						Core/Elective
U21CM3L1	PYTHON PROGRAMMING LAB						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Programming for Problem Solving Lab	-	-	-	3	25	50	1.5

Course Objectives:

Develop ability to

1. Elucidate problem solving through python programming.
2. Learn the use of functions in python programming.
3. Analyze solutions using Object Oriented concepts.
4. Design programs using File Handling Functions.
5. Create Mini-projects using various libraries.

Course Outcomes:

At the end of the course, student would be able to

1. Summarize the fundamental concepts of python programming.
2. Outline the control statements and functions by writing python program.
3. Demonstrate file handling operations and packages.
4. Interpret object oriented programming in python.
5. Apply the suitable libraries to solve simple problems.

List of Experiments:

1 a) Write a python program for PythonVariables,ExecutingPythonfromtheCommandLine,Editing PythonFiles,PythonReservedWords.

b) Write a python program to add two numbers.

c) Write a program to demonstrate different number data types in python.

d) Write a program to perform different arithmetic operations on numbers in python.

2) a) Write a python program to print a number is positive/negative using if-else.

b) Write a python program to find largest number among three numbers.

c) Write a Python program to swap two variables

d) Python Program to print all Prime Numbers in an Interval

3) a) Write a python program to check whether the given string is palindrome or not.

b) Write a program to create, concatenate and print a string and accessing substring from a given string.

c) Functions:PassingparameterstoaFunction,VariableNumberofArguments,Scope,and PassingFunctionstoaFunction.

4) a) Create a list and perform the following methods

1) insert() 2) remove() 3) append() 4) len() 5) pop() 6)clear()

b) Create a dictionary and apply the following methods

1) Print the dictionary items 2) access items 3) useget () 4) change values 5) use len()

c) Create a tuple and perform the following methods

1) Add items 2) len() 3) check for item in tuple 4)Access items

5) a) OOP concepts: Classes, File Organization, Special Methods, Inheritance, Polymorphism, Special Characters, Character Classes, Quantifiers, Dot Character, Greedy Matches, Matching at BeginningorEnd,MatchObjects,Compiling RegularExpressions.

b) Write a python Program to call data member and function using classes and objects

6) a) Write a program to double a given number and add two numbers using lambda()

b) Write a program for filter () to filter only even numbers from a given list.

c) Write a Python Program to Make a Simple Calculator.

7) a) Demonstrate a python code to print try, except and finally block statements

b) Write a python program to open and write “hello world” into a file and check the access permissions to that file?

c) Python program to sort the elements of an array in ascending order and Descending order

8) a) Write a python program to open a file and check what are the access permissions acquired by that file using os module.
b) Write a program to perform basic operations on random module.

9) Write a python program to practice some basic library modules
a) Numpy
b) SciPy

10) Introduction to basic concept of GUI Programming and Develop desktop based application with python basic Tkinter() Module.

11) Write a python program to create a package (college),subpackage (alldept),modules(it,cse) and create admin function to module.

12) Write a python program to create a package (Engg), sub package(years),modules (sem) and create staff and student function to module.

Suggested Readings:

1. Gerald J. Kowalski, Mark T. Maybury: Information Storage and Retrieval Systems: Theory and Implementation, Second Edition Kluwer Academic Publishers
2. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms, Prentice Hall, 1992.
3. Modern Information Retrival By Yates Pearson Education.
4. Information Storage & Retieval By Robert Korfhage – John Wiley & Sons

Course Code	Course Title						Core/Elective
U21CS3L2	C PROGRAMMING LAB						Core
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
--	L	T	D	P			
	-	-	-	2	50	-	0

Course Objectives:

1. To understand the fundamentals of programming in C Language.
2. To write, compile and debug programs in C.
3. To formulate solution to problems and implement in C.
4. To effectively choose programming components to solve computing problems

Course Outcomes:

On completion of this course, students are able to:

1. Choose appropriate data type for implementing programs in C language.
2. Design and implement modular programs involving input output operations, decision making and looping constructs.
3. Implement search and sort operations on arrays.
4. To decompose a problem into functions and to develop modular reusable code
5. Apply the concept of pointers for implementing programs on dynamic memory management and string handling with design and implement programs to store data in structures and files.

List of experiments:

1. Finding maximum and minimum of given set of numbers, finding roots of quadraticequation.
2. Sinx and Cosx values using series expansion.
3. Conversion of binary to decimal, octal, hexadecimal and vice-versa.
4. Generating Pascal triangle, pyramid of numbers.
5. Recursion:factorial,Fibonacci,GCD.
6. Matrix addition and multiplication using arrays.
7. Programs on pointers: pointer to arrays, pointer to functions.
8. Functions for string manipulations.
9. Programs on structures and unions.
10. File handling programs

Suggested Reading:

1. Byron Gottfried, “Programming with C”, Schaum’s outlines, 2nd Edition, TATA McGraw-Hill.
2. A.K.Sharma, “Computer Fundamentals and Programming in C”, 2nd Edition, University Press.
3. E Balaguruswamy, “Programming in ANSI C”, Tata McGraw-Hill Education, 2008.
4. Brian W. Kernighan and Dennis M. Ritchie, “the C Programming Language”, Prentice Hall of India, 1988.

Course Code	Course title						Core/Elective
U21EN1L1	EFFECTIVE COMMUNICATION SKILLS LAB COMMON TO ALL BRANCHES						Core
Prerequisites	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Basic English	-	-	-	3	25	50	1.5

Course Objectives:

To enhance the listening and speaking skills of students by:

1. Giving them sufficient practice in listening with comprehension
2. Providing them ample opportunities to improve their public speaking skills and soft skills
3. Training them in the use of correct pronunciation, stress and intonation
4. Sensitizing them to the use of verbal and non-verbal communication appropriate to the context
5. Encouraging them to learn the art of conversation to suit formal and informal situation
6. Preparing them to make formal presentations and face interviews

Course Outcomes:

After completing this course, the student will be able to:

1.Listen, understand and interpret formal and informal spoken language
2.Speak English with acceptable pronunciation, stress and intonation
3.Present themselves with confidence in formal situations
4.Participate in individual and group activities with relative ease
5.Use verbal and nonverbal communication while using soft skills and make formal presentations and face interviews

LIST OF ACTIVITIES

1. Listening for comprehension
2. Pronunciation, Intonation, Stress and Rhythm
3. Conversation Skills
4. Introducing Oneself and Others
5. Asking for and Giving information
6. Making Request and Responding to them Appropriately
7. Giving Instructions and Responding to them Appropriately
8. Making Formal Announcement and Emceeing
9. Group Discussion
10. Just A Minute (JAM)
11. Role Play
12. Debate
13. Public Speaking Skills and Body Language
14. Interviews
15. Formal Presentations

Suggested Readings:

1. Board of Editors. Language and Life Skills Approach. Orient Black Swan,2018
2. BalaSubramaniam, T.A. Text book of English Phonetics for Indian Students, Macmillan, 1981.
3. CIEFL, Exercises in Spoken English. PART-III, Oxford University Press.
4. Pillai, Radhakrishna G. Spoken English for You – Level II. Emerald Publisher, 8th Edition.2014.
5. Sethi, J. and PV Dhamija. A Course in Phonetics and Spoken English. Prentice, India Learning Private Limited, 2nd Edition.1999
6. Robert. M. Sherfield& et al. Developing Soft Skills. Pearson Education.4th Edition. 2009.