B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

Department of Mathematics

Syllabus

Semester: III			
Course: Fourier Transform, Fundamentals of logic and Linear Algebra **Course Code: 22MAI131 (Common to CSE, ISE, AIML)**			
L:T:P:J	**2:1:1:0**	**CIA**	**: 50**
Credits:	**03**	**SEA**	**: 50**
Hours:	**40**	**SEA Duration**	**: 03 Hours**

Course Learning Objectives: The students will be able to

1 Have an insight into Fourier series, Fourier transforms.

2 Develop knowledge of Fundamentals of logic and Relations, Vector Spaces & Linear Transformation arising in engineering

Module-1: Fourier Series & Fourier Transforms	**No. of hours**	**Blooms cognitive Levels**
Examples from Engineering field that require Fourier series and Fourier Transforms. **Fourier series**: Periodic functions, Introduction to Fourier Series, Dirichlet's condition. Problems on Fourier series over $(-l\,,\,l)$. **Fourier Transforms:** Introduction to infinite Fourier transform, Fourier sine and cosine transform and properties, problems on infinite Fourier transform, Discrete & Fast Fourier Transform. ***Lab Component:*** *Finding the Fourier series and Fourier Transform of a function*	**L : 04** **T : 04**	**Apply**
Module-2: Fundamentals of logic and Relations		
Examples from Engineering field that require Fundamentals of logic and Relations. **Fundamentals of logic:** Basic connectives and truth tables, logic equivalence - the laws of logic, logical implication- rules of inference **Relations:** First order linear recurrence relation, second order linear homogenous recurrence relation with constant coefficients. ***Lab Component:*** *Finding the solution of recurrence relation*	**L : 04** **T : 04**	**Apply**
Module-3: Vector Spaces		
Examples from Engineering field that require vector spaces Recap of system of linear homogenous and non-homogeneous equation and solution sets. Vector spaces, subspaces, linearly independent and dependent, Linear span of a set, basis and dimension, coordinate vectors. ***Lab Component:*** *problems on linearly independent and dependent, basis and dimension of a vector space.*	**L : 04** **T : 04**	**Apply**
Module-4:Linear Transformation		
Examples from Engineering field that require linear transformation. Linear transformations, algebra of linear transformations, representation of transformations by matrices, Non-singular linear transformation, Inverse of a linear transformation, Range space, Null space and problems on Rank-nullity theorem. ***Lab Component:*** *problems on Inverse of a linear transformation and Rank-nullity theorem*	**L : 04** **T : 04**	**Apply**
Module-5: Inner Product Spaces		
Examples from Engineering field that require Inner product spaces. Inner products Inner product spaces, Orthogonal set, orthogonal projections, orthonormal bases, Gram-Schmidt process, QR-factorization, Recap of Eigen values and Eigen vectors, problems on singular value decomposition. ***Lab Component:*** *Problems on QR-factorization and singular value decomposition*	**L : 04** **T : 04**	**Apply**

Course Outcomes: After completing the course, the students will be able to

CO 1: Apply Fourier series & Transform concepts in Data visualization and Cryptography.

CO 2: Communicate the basic concepts of logic and their relevance for computer science engineering.

CO 3: Apply the knowledge of vector spaces for solving problems in arising in engineering field

CO 4: Apply the knowledge of Linear transform for solving problems in arising in image processing

CO 5: Compute orthogonal and orthonormal bases vectors and decomposition of a symmetric matrix using standard technique.

Text Books:

1. E. Kreyszig: “Advanced Engineering Mathematics”, John Wiley & Sons,10th Edition(Reprint), 2016.
2. B. S. Grewal: “Higher Engineering Mathematics”, Khanna Publishers, 44th Ed., 2017.
3. H. K. Dass, “ Advanced Engineering Mathematics” S. Chand publication.

Reference Books:

1. C. Ray Wylie, Louis C. Barrett : “Advanced Engineering Mathematics", 6" Edition, 2. McGraw-Hill Book Co., New York, 1995.
2. James Stewart : “Calculus —Early Transcendentals”, Cengage Learning India Private Ltd., 2017.
3. B.V.Ramana: "Higher Engineering Mathematics" 11th Edition, Tata McGraw-Hill, 2010.
4. Srimanta Pal & Subobh C Bhunia: “Engineering Mathematics”, Oxford University Press, 3"Reprint, 2016.
5. David C. Lay, Steven R. Lay and J. J. McDonald “Linear Algebra and its applications”, 3rd Edition, Pearson Education Ltd., 2017.

Ralph P. Grimaldi, “ Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education 2004.

Web links and Video Lectures:

1. https://nptel.ac.in/courses/111106111
2. https://youtu.be/OynpZwylau8
3. https://archive.nptel.ac.in/courses/111/106/111106051/
4. https://www.youtube.com/watch?v=zvRdbPMEMUI
5. https://www.youtube.com/watch?v=cHNmT1-qurk
6. https://www.youtube.com/watch?v=ATqV_I8DCh0

PCC	**CIA**	**SEA**	**CIA (50)**			**SEA** **Conduction: 100 M** **Reduced to: 50 M**
				I	II	
Conduction	50	50	Written Test	50	50	Five questions with each of 20 marks (with internal choice). Student should answer one full question from each module
				Average of two tests – 25 Marks		
			Assignment	15		
			AAT	10		
					Total – 50 marks	**Total – 50 marks**

i) CIA: 50%

IA Test: 2 IA tests - Each of 50 Marks	Average of 2 tests – scaled down to 25 M
Assignment – Two assignments – one for 10 marks and another for 5 marks	15 Marks
Additional Assessment Tools (AAT) – Oral /Online Quizzes, Presentations, Group discussions, Case studies, Term Paper, Open ended experiments, Mini industrial/social/rural Projects, Two-minute video on latest topic, Short MOOC courses, Practical Orientation on Design thinking, creativity & Innovation, Participatory & Industry integrated learning, Practical activities, Problem solving exercises, Participation in seminars/academic events/symposia and any other activity	10 Marks
Total	**50 Marks**

ii) SEA : 50%

Theory Exam	5 questions to answer each of 20 Marks 2 questions from each module with internal choice Student should answer one full question from each module	20 M x 5 = **100 M reduced to 50 M**
	Total	**50 Marks**

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

Computer Organization

SEMESTER–III

Subject Code: 22ISE132	L:T:P:J: 3:0:0:0	CIE Marks:	50
Credits:	3	SEE Marks:	50
Total Number of Lecture Hours	40	SEE Duration:	3Hours

Course objectives:

- To understand the basic sub systems of a computer, their organization, structure and operation.
- Illustrate the concept of programs as sequences of machine instructions.
- Teach the concepts of Memory system and cache memory.
- Cultivate clear thinking in performing Arithmetic, Multiplication, division and Floating-point number operations in computer.
- Describe the working of pipelining and multiprocessor computer architecture.

Module1: Introduction	**Teaching Hours**	**Blooms cognitive Levels**
Basic Structure of Computers: Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions.	**8**	**Understand**
Module 2: Input / Output Organization		
Accessing I/O Devices, Interrupts – Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, USB.	**8**	**Understand**
Module 3: Memory System		
Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations.	**8**	**Analyze**
Module 4: MIPS Arithmetic operations		
MIPS Addressing for 32-Bit Immediates and Addresses, Parallelism and Instructions: Synchronization, Translating and Starting a Program, Addition and Subtraction (MIPS), Multiplication and Division (MIPS).	**8**	**Apply**
Module 5: Pipelining and Multiprocessors		
Pipelining: Basic concepts, Data Hazards, Instruction hazards, Basic processing unit: some fundamental concepts, execution of complete instruction, multi-bus organization, The structure of general-purposes multiprocessors, **Parallel Computer Architecture:** Processor Architecture and Technology Trends, Flynn's Taxonomy of Parallel Architectures.	**8**	**Apply**

Course outcomes: The students should be able to:

- CO 1: Ability to understand the abstraction of various components of a computer.
- CO 2: Ability to understand the functions of different sub systems, such as processor, Input/output, and memory.
- CO 3 : Analyze the concepts of Memory system and cache memory.
- CO 4 : Apply Arithmetic, Multiplication and division operations in computer.
- CO 5: Apply the working of pipelining and multiprocessor computer architecture

Reference Books:

1. Carl Hamacher, ZvonkoVranesic, SafwatZaky, Computer Organization, 5th Edition, Tata McGraw Hill, 2002.
2. Parallel Programming for Multicore and Cluster Systems, Thomas Rauber, Gudula Runger, 2nd Edition, Springer, 2013.
3. David A. Patterson and John L. Hennessey, “Computer organization and design, The Hardware/Software interface”, Morgan Kauffman / Elsevier, Fourth/Fifth edition, 2014.
4. William Stallings: Computer Organization & Architecture, 9th Edition, Pearson, 2015.

PCC	**CIA**	**SEA**	**CIA (50)**			**SEA** **Conduction: 100 M** **Reduced to: 50 M**
				I	II	
Conduction	50	50	Written Test	50	50	Five questions with each of 20 marks (with internal choice). Student should answer one full question from each module
				Average of two tests – 25 Marks		
			Assignment	15		
			AAT	10		
					Total – 50 marks	**Total – 50 marks**

i) CIA: 50%

IA Test: 2 IA tests - Each of 50 Marks	Average of 2 tests – scaled down to 25 M
Assignment – Two assignments – one for 10 marks and another for 5 marks	15 Marks
Additional Assessment Tools (AAT) – Oral /Online Quizzes, Presentations, Group discussions, Case studies, Term Paper, Open ended experiments, Mini industrial/social/rural Projects, Two-minute video on latest topic, Short MOOC courses, Practical Orientation on Design thinking, creativity & Innovation, Participatory & Industry integrated learning, Practical activities, Problem solving exercises, Participation in seminars/academic events/symposia and any other activity	10 Marks
Total	**50 Marks**

ii) SEA : 50%

Theory Exam	5 questions to answer each of 20 Marks 2 questions from each module with internal choice Student should answer one full question from each module	20 M x 5 = **100 M reduced to 50 M**
	Total	**50 Marks**

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

Operating Systems

SEMESTER–III

Subject Code: 22ISE133	L:T:P:J: 3:0:0:0	CIA Marks:	50
Credits:	3	SEA Marks:	50
Total Number of Lecture Hours	40	SEA Duration:	3 Hours

Course Objectives:

- Introduce concepts and terminology used in OS
- Explain threading and multithreaded systems
- Illustrate process synchronization and concept of Deadlock
- Introduce to Unix File Systems

Module1: Introduction to Operating System & Process Management	**Teaching Hours**	**Blooms cognitive Levels**
Fundamental Concepts of Operating System: Introduction to Operating systems, Operating system functions and services, System boot. **Process Management:** Process abstraction, process address space, process management, system calls, threads. **CPU Scheduling:** Levels of scheduling, comparative study of scheduling algorithms, Multilevel Queue Scheduling, Multi-processor scheduling.	**8**	**Apply**
Module2: Process Synchronization and Deadlocks		
Concurrent Processes: Critical section problem, semaphores, Classical problems of synchronization, monitors, inter-process communication, message passing mechanisms. **Deadlocks:** Characterization, prevention and avoidance, deadlock detection and recovery.	**8**	**Apply**
Module3: Memory Management		
Memory Management: Background; Swapping; Paging; Structure of page table; Segmentation, virtual memory concept, demand paging, page replacement algorithms, thrashing, Disk Scheduling. **Solid State Drives**-SSD Architecture, Flash Controller,	**8**	**Apply**
Module4: UNIX file system		
Unix files: UNIX Architecture, Naming files. Basic file types/categories. Organization of files. Hidden files. Standard directories. Parent child relationship. The home directory and the HOME variable. Reaching required files- the PATH variable, manipulating the PATH, Relative and absolute pathnames. Directory commands – pwd, cd, mkdir, rmdir commands. The dot (.) and double dots (..) notations to represent present and parent directories and their usage in relative path names. File related commands – cat, mv, rm, cp, wc and od commands. **Practical component**: Execution of UNIX Shell Commands	**8**	**Apply**
Module5: File attributes and Shell programming		
File attributes and permissions: The ls command with options. Changing file permissions: the relative and absolute permissions changing methods. Recursively changing file permissions. Directory permissions. The shells interpretive cycle: Wild cards. Removing the special meanings of wild cards. Three standard files and redirection. **Connecting commands**: Pipe, grep, egrep. **Shell programming**: Ordinary and environment variables. Read and read-only commands. Command line arguments. exit and exit status of a command. Logical operators for conditional execution. The test command and its shortcut. The if, while, for and case control statements. The set and shift commands and handling positional parameters. The here (<<) document. Simple shell program examples. **Practical component:** Execution of Wildcards & UNIX Shell Programs	8	**Analyze**

Course Outcomes: After completing the course, the students will be able to	
CO 1	Apply the concepts of process scheduling to improve CPU utilization and identify various multi-threading models
CO 2	Identify the need of policies, protection required in managing deadlock, main and virtual memory & various techniques in managing concurrent processes
CO 3	Apply the concept of paging & segmentation for effective memory management
CO 4	Apply the concepts of Unix system and file commands to perform various tasks in files and system.
CO 5	Analyze the concepts of Wildcards and Shell Programming to write basic shell scripts and formulating regular expressions for Pattern matching

Reference Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 7th edition, Wiley-India, 2006
2. Sumitabha Das., Unix Concepts and Applications., 4thEdition., Tata McGraw Hill.
3. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 9th Edition, 2018.
4. W. Richard Stevens: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education, 2005
5. Unix System Programming Using C++ - Terrence Chan, PHI, 1999.

Web links and Video Lectures:

1. https://academicearth.org/
2. https://archive.nptel.ac.in/courses/106/105/106105214/

PCC	CIA	SEA	CIA (50)			SEA Conduction: 100 M Reduced to: 50 M
				I	II	
Conduction	50	50	Written Test	50	50	Five questions with each of 20 marks (with internal choice). Student should answer one full question from each module
				Average of two tests – 25 Marks		
			Assignment	15		
			AAT	10		
					Total – 50 marks	**Total – 50 marks**

i) CIA: 50%

IA Test: 2 IA tests - Each of 50 Marks	Average of 2 tests – scaled down to 25 M
Assignment – Two assignments – one for 10 marks and another for 5 marks	15 Marks
Additional Assessment Tools (AAT) – Oral /Online Quizzes, Presentations, Group discussions, Case studies, Term Paper, Open ended experiments, Mini industrial/social/rural Projects, Two-minute video on latest topic, Short MOOC courses, Practical Orientation on Design thinking, creativity & Innovation, Participatory & Industry integrated learning, Practical activities, Problem solving exercises, Participation in seminars/academic events/symposia and any other activity	10 Marks
Total	**50 Marks**

ii) SEA : 50%

Theory Exam	5 questions to answer each of 20 Marks 2 questions from each module with internal choice Student should answer one full question from each module	20 M x 5 = **100 M reduced to 50 M**
	Total	**50 Marks**

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering

Data Structures and its Applications

SEMESTER– III

Subject Code: 22ISE134	L:T:P:J: 3:0:2:0	CIE Marks:	50
Credits:	4	SEE Marks:	50
Total Number of Lecture Hours	50	SEE Duration:	3 Hours

Course objectives:

- Learn the fundamental data structures and identify data structuring strategies that are appropriate to a given contextual problem and able to design, develop, test and debug in C language considering appropriate data structure. Use and implement structures to obtain solutions.
- Illustrate and implement data types such as stack, queue and linked list and apply them for the given problem.
- Understand and distinguish the conceptual and applicative differences in trees, binary trees and binary search trees. Apply the correct tree for the given application.
- Create and use appropriate data structures in C programs for solving real life problems.

Module 1: Stacks and Queues	**Teaching Hours**	**Blooms cognitive Levels**
Concept Learning: Data Structures: Classification (Primitive & Non-primitive), Operations,Pattern Matching Algorithms (Brute force, KMP) **Stacks:** Definition, Operations, Implementation using arrays, Applications of Stacks – Infix toPostfix Conversion and Postfix Expression Evaluation. **Queues:** Definition, Operations, Implementation, Applications, Circular Queue (Message queueusing Circular queue), Doubly Ended Queue, Priority Queue. **Practical session:** Tracing Exercises, Analysis Exercises, Exercises to Modify programs **Programs not limited to:** 1. Implement KMP pattern matching algorithm for a given main string and pattern 2. Implement Boyer-Moore pattern matching algorithm for a given main string and pattern 3. Program to implement stack operations (push, pop, and display) using static array and also on dynamic array and compare the performance 4. Program to convert infix expression into postfix. 5. Program to evaluate given postfix expression 6. Program to implement queue operations using static arrays and on dynamic arrays and compare the performance. 7. Program to implement circular queue operations using arrays and on dynamic arrays and compare the performance	**10**	**Apply**
Module 2: Linked List		
Linked Lists: Definition, Create, Insert, Delete, Update, Traverse, and Position-based Operations, Concatenate, Merge, and Reverse Lists, Doubly Linked List Implementation and Operations, Circular Linked List Implementation and Operations, Applications of Lists (Polynomial addition). Implementation of stacks and queues using Linked List **Practical session:** Tracing Exercises, Analysis Exercises, Exercises to Modify programs **Programs not limited to:** 1 Program to create single linked list and implement its operations with and without header nodes I. Insert (front and rear end) ii. Delete. (Front and rear end) iii. Search. iv. Reverse. 2 Implement DLL as a stack and also as a queue	**10**	**Apply**

3 Create 2 CLL of sparse polynomials and perform addition on these 2 polynomials. Represent the resulting polynomial in a CLL 4 Implement multiple stacks and multiple queues in a singly linked list		
Module 3: Trees		
Concept Learning: Trees: General Tree Representation, Traversals, Applications. **Binary Trees**: Definition, Properties, Traversals, Applications. **Binary Search Tree**: Definition, Implementation, Search, Insert, Delete operations. Building andEvaluating Binary Expression Tree. **Concept Learning Graphs:** Adjacency Lists, and Traversal of Graphs (BFS and DFS) **Practical session:** Tracing Exercises, Analysis Exercises, Exercises to Modify programs **Programs not limited to:** 1 Implement the following operations on Binary tree: - A. Count the number of nodes in the binary tree B. Count the leaf nodes C. Check if 2 trees are equal D. Perform Tree Traversals namely in-order, pre-order, post-order and level order. 2 Create a Binary search Tree for a set of integer values, perform deletion of a key and searching for a key 3 For a given input graph check weather the nodes are connected or not using BFS and DFS method 4 Create a expression tree to evaluate the given expression using binary tree	**10**	**Apply**
Module 4: Advanced Trees & Hashing		
Concept Learning: Heap Tree: Definition, Implementation, Insert, Delete, Peek operations.**Hashing**: Hash Table, Hash Functions, Collision Handling by Open Addressing, Chaining. **Practical session:** Tracing Exercises, Analysis Exercises, Exercises to Modify programs. **Programs not limited to:** 1 Create a max heap and min heap of integers, display the values and perform deletion operations 2 Program to implement hashing with collision resolution	**10**	**Apply**
Module 5: File Structures		
File Structures: Primary Indexing, Secondary Indexing, B-Trees, Co sequential processing, K way merge. **Practical session:** Tracing Exercises, Analysis Exercises, Exercises to Modify programs. **Programs not limited to:** 1 Write a program to implement simple index on primary key for a file. Implement add (), search (), delete () using the index. 2 Write a program to read two lists of names and then match the names in the two lists using Consequential Match based on a single loop. Output the names common to both the lists. 3 Write a program to read k Lists of names and merge them using k-way merge algorithm with k = 8.	**10**	**Apply**

Course outcomes: The students should be able to:
CO1: Apply fundamental data structures viz., Lists, Stacks, Queues, Linked Lists, Binary Trees from first principles CO2: Identify the use of appropriate data structures for a given problem CO3: Design and implement solutions to basic practical problems using customized data structures CO4: Apply the Advanced concepts like Heap & Hashing to solve problems. CO5: Apply the concepts to solve graphical problems.

Reference Book:

1. Fundamentals of Data Structures in C -- by Horowitz, Sahni and Anderson-Freed (Silicon Press 2007).
2. Data Structures Using C, Reema Thareja, 1st Edition, 2011, Oxford Higher Education, ISBN-13: 978-0198099307
3. “Data Structures and Program Design in C”, Robert Kruse, C L Tondo, Bruce Leung and ShashiMogalla, PHI, 2nd Edition, 2015.
4. Y. Langasm, M. J. Augenstein, A. M. Tenenbaum (2001) Data Structures Using C and C++, PrenticeHall India, New Delhi, India.
5. T. H. Cormen, C. E. Leiserson and R. L. Rivest (1990) Introduction to Algorithms, Third Edition, MITPress, MA.
6. Data Structures and Algorithm Analysis in C++, Mark Allen Weiss, 4th Revised edition; 2013,Addison-Wesley, ISBN-13: 978-8131714744

PCI	**CIA**	**SEA**	**CIA (50)**			**SEA** **Conduction: 100 M** **Reduced to: 50 M**
				I	II	
Conduction	50	50	Written Test	50	50	Five questions with each of 20 marks (with internal choice). Student should answer one full question from each module
				Average of two tests – 50 marks scaled down to 15 marks		
			Assignment	Average of 2 Assignments – 10M		
			Practical	Weekly Assessment – 10 Marks IA test – 15 Marks (IA test to be conducted for 50 M and scaled down to 15M)		
			Total – 50 Marks			**Total – 50 Marks**

i) CIA: 50%

Theory	**IA Test** (Theory): 2 IA tests - each of 50 Marks – Average of 2 tests scaled down to 15 Marks **Assignment** : 2 Assignments – each of 10 marks	25 Marks
	Weekly Assessment – 10 Marks **Practical test (1)** - 15 marks	25 Marks
	Total	**50 Marks**

ii) SEA : 50%

Question Paper:

Theory Exam	5 questions to answer, each of 20 Marks 2 questions from each module with internal choice Student should answer one full question from each module	20 M x 5 = **100 M** **Reduced to 50 M**
	Total	**50 Marks**

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering

Object Oriented Programming using Java
III Semester

Course Code: 22ISE135	**L:T:P:J 3:0:2:0**	**CIA Marks**	**50**
Credits	**04**	**SEA Marks**	**50**
Total Number of Contact Hours	**50**	**SEA Duration**	**03**

Course Learning Objectives: This course will enable students to:

- Show competence in the use of the Java programming language in the development of small to medium-sized application programs that demonstrate professionally acceptable coding and performance standard
- Understand the basic principles of the object-oriented programming
- Demonstrate an introductory understanding of graphical user interfaces, multithreaded programming, and event-driven programming.

Module – 1	**No. of hours**	**Blooms cognitive Levels**
Basics of Java: Features of Java, Byte Code and Java Virtual Machine, JDK, Data types, Operator, Control Statements – If , else, nested if, if-else ladders, Switch, while, do-while, for, for-each, break, continue. **Array and String:** Single and Multidimensional Array, String class, StringBuffer class, Operations on string, Command line argument, Use of Wrapper Class. **Classes, Objects and Methods:** Class, Object, Object reference, Constructor, Constructor Overloading, Method Overloading, Recursion, Passing and Returning object form Method, new operator, this and static keyword, finalize() method **Programs:** 1. Write a program to convert rupees to dollar. 60 rupees=1 dollar. 2. Write a program that calculate percentage marks of the student if marks of 6 subjects are given. 3. Write a program to enter two numbers and perform mathematical operations on them. 4. Write a program to find length of string and print second half of the string. 5. Write a program to accept a line and check how many consonants and vowels are there in line. 6. Write a program to count the number of words that start with capital letters. 7. Write a program to find that given number or string is palindrome or not. 8. Create a class which asks the user to enter a sentence, and it should display count of each vowel type in the sentence. The program	**12**	**Apply**

should continue till user enters a word “quit”. Display the total count of each vowel for all sentences. 9. Write an interactive program to print a string entered in a pyramid form. 10. Write an interactive program to print a diamond shape. 11. Program to define class Box and constructors. Demonstrate constructors. 12. Program to define class Addition, methods and objects. Demonstrate method overloading to add 2 integers, 3 floating numbers and 2 strings.		
Module – 2		
Class and Objects (Contd.) Access control, modifiers, Nested class, Inner class, Anonymous inner class, Abstract class. **Inheritance:** Use of Inheritance, Inheriting Data members and Methods, constructor in inheritance, Multilevel Inheritance – method overriding Handle multilevel constructors – super keyword, Stop Inheritance - Final keywords, Dynamic method dispatch, Understanding of Java Object Class, **Package and Interfaces:** Use of Package, CLASSPATH, Import statement, Static import, Access control Creation and Implementation of an interface, Interface reference, Interface inheritance, Nesting Interfaces, variables in Interfaces, Comparison between Abstract Class and Interface **Programs:** 1. Program to define inheritance and show method overriding - Program to define base class Shapes, with methods and create derived classes Square, Triangle, Rectangle and Circle. Create objects and show method overriding for calculating area of Square, Triangle, Rectangle and Circle. 2. Create an Interface Stack with methods push(), pop() and show(). Create class IntStack which implements Stack. Demonstrate the working of this class. 3. Create an Interface Queue with methods insert(), delete() and display(). Create class StringQueue which implements Queue. Demonstrate the working of this class	10	**Apply**
Module – 3		
Exception Handling: Exception and Error, Use of try, catch, throw, throws and finally, Built in Exception, Custom exception, Throwable Class. **Multithreaded Programming**: Use of Multithread programming, Thread class and Runnable interface , Thread priority, Thread synchronization, Thread communication, Deadlock **Programs:** 1. Analyze how a programmer must handle an error when an invalid mobile number is entered (lesser than 10 digits or greater than 10	**10**	**Apply**

digits; with alphabets; starting with 0) is entered while taking the details of a student. The Student class must contain the following: Name; USN; Branch and Mobile Number. Write a program using a custom exception type 2. Create an exception called Non Int Result Exception, which is generated when the result of dividing two integer values produces a result with a fractional component. Non Int Result Exception has two fields that hold integer values; a constructor; and an override of the to String() method, allowing a friendlier description of the exception to be displayed using println(). 3. Write a JAVA program which creates two threads, one thread displays "JAVA" 10 times, and another thread displays "PROGRAMMING LANGUAGE" 15 times continuously? 4. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number 5. Write a program that prints the Multiplication Table for numbers 2-10. At a time the tables must print completely for one number before printing for the next number 6. Create a Counter class with a private count instance variable and two methods. The first method: synchronized void increment() – tries to increment count by 1. If count is already at its maximum of 3, then it waits until count is less than 3 before incrementing it. The other method is synchronized void decrement() – attempts to decrement count by 1. If count already at a minimum of 0, then it waits until count is greater than 0 before decrementing it. Every time either method has to wait, it displays a statement as to why it is waiting. Also every time an increment or decrement occurs, the Counter displays a statement that says what occurred and shows count's new value. 7. Create a Bank class with withdrawal and deposit methods. Demonstrate the use of notify() and wait().		
Module – 4		
IO Programming: Introduction to Stream, Byte Stream, Character stream, Readers and Writers, File Class, File Input Stream, File Output Stream, Input Stream Reader **Swings:** Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application; Create a Swing J Applet; J label and Image Icon; J Text Field; The Swing Buttons; J Tabbed pane; J Scroll Pane; JList; J Combo Box; JTable. **Programs:**	**10**	**Apply**

1. Create a class called Student. Write a student manager program to manipulate the student information from files by using File Input Stream and File Output Stream 2. Refine the student manager program to manipulate the student information from files by using the Buffered Reader and Buffered Writer 3. Refine the student manager program to manipulate the student information from files by using the Data Input Stream and Data Output Stream. Assume suitable data 4. Create a GUI for a business of your choice using swings. Demonstrate the use of Button, Radio Button, Combo box, List, Text field etc.		
Module – 5		
Servlet : The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The Javax. servlet Package; Reading Servlet Parameter; The Javax. servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking. Java Server Pages (JSP): JSP, JSP Tags, Tomcat, Request String, User Sessions, Cookies, Session Objects **Programs:** 1. Write Servlet application to print current date &time 2. Write Servlet application to demonstrate session tracking 3. Implement Program 1 using JSP 4. Implement Program 2 using JSP	**8**	**Apply**
Course Outcomes: After completing the course, the students will be able to		
CO 1**:**Understand object-oriented programming concepts and implement in java. CO 2**:**Comprehend building blocks of OOPs language, inheritance, package and interfaces. CO 3**:**Identify exception handling methods. CO 4**:**Implement multithreading in object oriented programs CO 5**:**Describe how servlets fit into Java-based web application architecture		
Reference Books : 1. The Complete Reference, Java 2 (Eleventh Edition), Herbert Schild, - TMH. 2. Java Fundamentals A comprehensive introduction By Herbert Schildt, Dale Skrien, McGraw Hill Education. 3. Programming with Java A Primer – E.Balaguruswamy,McGrawhill 4. Core Java Volume-I Fundamentals Horstmann& Cornell, - Pearson Education. - Eleventh Edition **5.** Head First Java: A Brain-Friendly Guide, 3rd Edition- Kathy Sierra, Bert Bates		

PCI	**CIA**	**SEA**	**CIA (50)**			**SEA** **Conduction: 100 M** **Reduced to: 50 M**	
				I	II	**PART A**	**PART B**
Conduction	50	50	IA Test	30	30	30 Marks	70 Marks
				Average of two tests – 30 M			
			Continuous Assessment	Weekly Assessment -20 marks			
					Total – 50 Marks		**Total – 50 Marks**

i) CIA: 50%

IA Test: 2 IA tests - each of 30 Marks	Average of 2 tests – 30 M
Practical Lab record – 10 Marks Performance – 05 Marks Viva – 05 Marks	20 Marks
	Total 50 Marks

ii) SEA : 50%

Question Paper:

Theory part	5 questions to answer each of 6 Marks 2 questions from each module with internal choice Student should answer one full question from each module	6 M x 5 = 30 Marks
Execution part	Write up - 20 Marks Conduction - 40 Marks Viva-Voce - 10 Marks	70 Marks
	Total	**100 Marks** **Reduced to 50 M**

Note:

- No Assignment and AAT

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering

WEB TECHNOLOGIES
SEMESTER – III

Course Code: 22ISE136	**L:T:P:J 0:0:2:2**	**CIA: 50**
Credits:	**02**	**SEA: 50**
Hours:	**25**	**SEE Duration:03**

Course Learning Objectives: This course will enable students to:

1. Learn about HTML, JavaScript
2. Learn about Scopes and Closures in java Script, develop a Server-side programming using java Script, Node.js
3. Demonstrate a connectivity with Database MySQL using ODBC and Node.js
4. Implement CRUD operations in SQL using Node.js
5. Develop a client-side Application using React Js
6. Implement MERN Stack.

Programs List: (Not restricted to the list given)

PART -A

Module – 1	**Teaching Hours**
Introduction to HTML, what is HTML and Where did it come from? HTML Syntax, Semantic Markup, Structure of HTML Documents, HTML Elements, HTML5 Semantic Structure Elements. HTML Tables and Forms, Introducing Tables, Styling Tables, Introducing Forms, Form Control Elements, Table and Form Accessibility. **List of Programs** 1. Design a web page that shows your class timetable using HTML table tag. 2. Design a form which should accept the student data such as, USN, Name of the Student, Date of birth, Branch, Semester. After submitting the form, It should display confirmation message.	**5**
Module – 2	
JavaScript: What is JavaScript and What can it do? JavaScript Design Principles, where does JavaScript Go? Syntax, JavaScript Objects, The Document Object Model (DOM), JavaScript Data types and variables, Operators, screen output and keyboard input, Control statements. **List of Programs** 1. Write a JavaScript to design a simple calculator to perform the following operations: Sum, product, difference, and quotient. 2. Write a JavaScript that calculates the squares and cubes of the numbers from 0 to 10 and outputs HTML text that displays the resulting values in an HTML table format.	**5**
Module-3	
Javascript Functions, Recursion, Arrays, Array Methods, Strings, String Methods, Regular expressions, JavaScript Events, Validating form Input, OOP concepts in Javascript.	**5**

List of Programs 1. . Develop and demonstrate a HTML5 file that includes JavaScript script that uses functions for the following problems: a. To check whether a string is palindrome or not. b. Fibonacci Sequence using Recursion. 2. Create a student registration form for job application and validate the form fields using JavaScript. 3. Develop a JavaScript program to shuffle deck of cards.	
Module-4	
Introduction to node.js, node.js modules, NPM, node.js HTTP module, node.js file system, node.js email. Introduction to MySQL, Database creation in MySQL, Insertion in MySQL, Queries to perform CRUD operations using SQL, ODBC connectivity with MySQL. **List of Programs** 1. Develop a Server-side programming using java Script with the help of Node.js to perform following operations: 1. Access/Write a file on server. 2. Process User Input 2. Write a Program to implement CRUD Operation in SQL using Node.js 3. Demonstrate Open Database Connectivity (ODBC) with MySQL.	**5**
Module-5	
Introduction to React.js, react render HTML, How to create a new react application, react components, react class, react events, react props, react conditionals, react lists, react forms, react router, react CSS styling. **List of Programs** 1. Create a to-do list application with add and delete functionality. 2. Create a form that takes in a name and email address and displays the entered data below the form. **Self Study:** Write a Java Script Program to implement MERN Stack.	**5**
PART - B (Mini - Project) Not restricted to the list given	
Develop a web application project using the languages and concepts learnt in exercises listed in part A with a good look and feel effects. You can use any web technologies and frameworks and databases. Sample Projects Include : 1. Food Ordering Website 2. Online Purchase Store 3. Search Engine etc. 4. College website 5. Banking application 6. Blood donation application 7. Gaming application 8. Library management system 9. Chat application 10. Tourism website	**25**

Course outcomes: The students will be able to:	
CO1	Create web pages with various media contents using HTML5.
CO2	Create a robust Client-side validation with java script
CO3	Create an application using node.js and react js
Text Books :	
1. Programming the World Wide Web, 7th edition, Robert W. Sebesta , Pearson Education, ISBN- 9789332518827. 2. Get Programming with Node.js, Jonathan Wexler 1st Edition 3. Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, Krupa Chinnathambi, (2nd Edition)	
Reference Book:	
1. Web Application Design and Implementation: Apache 2, PHP5, MySQL, JavaScript, and Linux/UNIX Steven A. Gabarro, December 2006, c2007, Wiley-IEEE Computer Society Press. 2.Nate Murray, Felipe Coury, Ari Lerner and Carlos Taborda, “ng-book, The Complete Book on Angular 4” September 2016 3. KrasimirTsonev, “Node.js by Example Paperback”, May 2015. 3. Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development”, 1st Edition, Pearson Education India. (ISBN:978-9332575271).	
WEB links and Video Lectures	
1. https://www.w3schools.com/ 2. https://nodejs.org/en/ 3. https://www.mongodb.com	

PBL	**CIA**	**SEA**	**CIA (50)**			**SEA** **Conduction: 100 M** **Reduced to: 50 M**
Conduction	50	50	Theory	**I IA**	**II IA**	Project Assessed for 100 marks reduced to 50 Marks
				25	25	
				Average of 2 tests – 25 M		
			Practical	Weekly Assessment (Record/Project) – 10 Marks Lab IA test – 15 Marks		
					Total – 50 Marks	**Total – 50 Marks**

i) CIA: 50%

Theory - 2 IA tests - Each of 25 Marks	25 Marks
Practical **Weekly Assessment** - Lab record/Project – 10 Marks **Lab IA test** – 15 Marks	25 Marks
Total	**50 Marks**

ii) SEA : 50%

Project	Write up – 10 Marks Project report – 25 Marks Presentation & Demonstration - 50 Marks Viva-Voce – 15 Marks	100 Marks Reduced to 50 Marks

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Semester: III		
COURSE: Soft Skill-1		
Course Code: 22SFT138	**L:T:P:J: 0:0:2:0**	**CIA Marks:** 50
Credits:	**1**	**SEA Marks: 5 0**
Hours:	**24 hrs**	**SEA Duration: 2Hrs**

Course Learning Objectives: The students will be able	
1	To help students understand their strengths and weakness.
2	To develop analytical and creative ability to solve problems individually or as a team.
3	To make students industry ready through practice of corporate etiquettes.
4	To enhance public speaking and presentation skills.

Module No.	Contents of the Module	Hours	Cos
1	**Module-1 Understanding and Managing Self** Self-Awareness, Self-Management, Anger Management, Time management, Change management. Vision and goal setting - Diff between vision and goal, smart, stretched goal concept, case studies Knowledge, Skill, Attitude Personality analysis using Big 5 personality test Critical Thinking, Problem solving, Creativity and innovation Integrity, ethics, values	8	**1 &2**
2	**Module -2 Corporate etiquettes and Mannerism** Introduction to Etiquette and Mannerism, Personal Etiquette, Grooming etiquettes- professional styling, Body & personality styling, Video Interview Etiquettes, Personal Interview EtiquettesEffective meeting skills. Workplace behavior, Personal interview	6	**3**
3	**Module -3 Public Speaking and presentation skills** Introduction to public speaking, making ideas, illustrating and delivering ideas, overcoming fear of public speaking and developing great delivery. Advanced Business presentation skills, PowerPoint presentation, Group discussion	**6**	**4**
4	**Module -4 Team Work** Interpersonal skills, group work vs team work	**4**	**5**

Course Outcomes: At the end of the course the student will be able to:

CO1	Understand their strength and weaknesses.
CO2	Develop analytical and creative ability to solve problems.
CO3	Identify themselves as industry ready through the practice of corporate etiquettes.
CO4	Enhance public speaking and presentation skills.
CO5	Build team collaboration by working towards shared goals.

Mapping of Course Outcomes with Programme Outcomes:

COs	PO8	PO9	PO10	PO11
CO1	3	3		
CO2		3		3
CO3	3	3		3
CO4		3	3	
CO5			3	3

MOOC Course:

Communicate with impact - https://www.coursera.org/learn/communicate-with-impact

Leading Diverse Teams - https://www.coursera.org/learn/leading-diverse-teams

Practical component:

1. Mock GD and interview may be conducted at the end of the course to check their confidence. Students can prepare their SWOT analysis and present the same.
2. The students are to be involved in various activities and games such as Just a Minute or Pick and speak to demonstrate each topic.

Class Internal Assessment – 50 Marks

1. **Video Assignment -30Marks**
2. **Weekly Assessment -20Marks**

Rubrics for evaluation: (TOTAL - 30 Marks)

SL no.	Assessment	COs	Marks
1	Creativity	CO 2	5M
2	Approach and flow	CO 2	5M
3	Time Management (duration of video and deadline)	CO 1	5M
4	Individual presentation in the video	CO 4	5M
5	Report- Brief about the topic and Contribution of team members	CO 5	5M
6	Report- Reflections (learnings from the activity)	CO 2 & CO 5	5M

Semester End Assessment – 50 Marks

PPT	**- 10 Marks**
Communication (Clarity and English)	**- 10 Marks**
Body Language	**- 10 Marks**
Viva (Q and A)	**- 10 Marks**
Project Report	**- 10 Marks**

Final Marks = CIA + SEA = 50+50 = 100 Marks

B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

Department of Mathematics

Syllabus

Semester: IV		
Course: Statistics, Probability and Graph theory **Course Code: 22MAI141 (Common to CSE, ISE, AIML)**		
L:T:P:J	**2:1:1:0**	**CIA: 50**
Credits:	**03**	**SEA: 50**
Hours:	**40**	**SEA Duration: 03 Hours**

Course Learning Objectives: The students will be able to	
1	Provide an insight into applications of Graph Theory, Curve fitting & Statistical methods.
2	Develop the knowledge of probability, joint probability distribution and Queuing theory occurring in digital signal processing, design engineering and micro wave engineering.

Module-1: Curve fitting & Statistical methods	**No. of hours**	**Blooms cognitive Levels**
Examples from Engineering field that require curve fitting and statistical methods. **Curve Fitting:** Curve fitting by the method of least squares-fitting the curves of the form: $y = ax+b$, $y = ax^b$ and $y = ax^2 + bx + c$. **Statistical methods:** Introduction to Moments, Skewness, kurtosis and problems. Karl Pearson's coefficient of correlation and lines of regression. ***Lab Component:*** *Problems on curve fitting and statistical methods*	**L: 04** **T: 04**	**Apply**
Module-2: Probability Distributions & Joint probability distribution		
Examples from Engineering field that require Probability and Joint probability distribution **Probability Distributions:** Review of basic probability theory. Discrete and continuous Random variables, probability mass/density functions (definitions only). Binomial, Poisson, exponential and normal distributions (without proof). **Joint probability distribution:** Joint Probability distribution for two discrete random variables, expectation, covariance and correlation. ***Lab Component:*** *Problems on binomial, Poisson, exponential and normal distributions*	**L: 04** **T: 04**	**Apply**
Module-3:Markov Chain & Sampling Theory		
Examples from Engineering field that require Markov Chain and Sampling Theory **Markov Chain**: Introduction to Stochastic process, Probability vectors, Stochastic matrices, Regular stochastic matrices, Markov Chains, Higher transition probabilities, Stationary distribution of Regular Markov chains and absorbing states, Markovian processes. **Sampling Theory**: Introduction to sampling theory, Testing of hypothesis, level of significance, confidence limits, test of significance of mean and difference of means for large samples-z-test, test of significance of small Samples-Student's t- distribution, Goodness of fit-Chi-Square test. ***Lab Component:*** *Problems on Markovian processes and, Sampling Theory*	**L: 04** **T: 04**	**Apply**
Module-4: Queuing theory		
Examples from Engineering field that require queueing theory Introduction, Symbolic representation of a queuing model, single server Poisson queuing model with infinite capacity (M/M/1: ∞/FCFS), when $\lambda_n = \lambda$ and $\mu_n = \mu(\lambda < \mu)$, Multiple server Poisson queuing model with infinite capacity (M/M/S: ∞/ FCFS), when $\lambda_n = \lambda$ for all $n, (\lambda > S\mu)$, ***Lab Component:*** *Problems on* (M/M/1: ∞/FCFS) and (M/M/S: ∞/ FCFS) queuing models	**L: 04** **T: 04**	**Apply**
Module-5: Graph Theory		
Examples from Engineering field that require graph theory Basic concepts, types of graphs, order and size of a graph, in-degree and out-degree, bipartite-graphs, connected and disconnected graphs, Eulerian graph, Hamiltonian graphs, sub-graphs, isomorphic graphs. Matrix representation of graphs, adjacency matrix, incidence matrix. Planar graphs: definition, characterization of planar graphs, Kuratowski's theorem, Euler's formula and consequences. ***Lab Component:*** *Problems on detection of planar and non-planar graphs*	**L: 04** **T: 04**	**Apply**

Course Outcomes: After completing the course, the students will be able to

CO 1: Make use of correlation and regression analysis to fit a suitable mathematical model for the statistical data.

CO 2: Apply discrete and continuous probability and joint probability distributions in analyzing the probability models arising in engineering field.

CO 3: Use Markov chain in prediction of future events and demonstrate the validity of testing the hypothesis.

CO 4: Acquire skills in analyzing queuing models.

CO 5: Apply the knowledge of Graph Theory in Network modeling, electrical network and computational algorithms

Reference Books:

1. E. Kreyszig: "Advanced Engineering Mathematics", John Wiley & Sons, 10th Edition(Reprint), 2016.
2. B. S. Grewal: "Higher Engineering Mathematics", Khanna Publishers, 44th Edition, 2017.
3. S. D. Sharma : "Operations Research", Kedar Nath Ram Nath & Co. Meerut, 2014.
4. T. Veerarajan : Probability, Statistics and Random processes, McGraw Hill Education(India) Private Limited, Third edition, Nineteenth reprint 2017.
5. C. Ray Wylie, Louis C. Barrett : "Advanced Engineering Mathematics", 6th Edition, McGraw-Hill Book Co., New York, 1995.
6. James Stewart : Calculus-Early Transcendental, Cengage Learning India Private Ltd., 2017.
7. B. V. Ramana: "Higher Engineering Mathematics" 11th Edition, Tata McGraw-Hill, 2010.
8. Srimanta Pal & Subobh C. Bhunia: "Engineering Mathematics", Oxford University Press, 3rd Reprint, 2016.
9. Narsingh Deo, "Graph Theory with Applications to Engineering and Computer Science", Prentice Hall of India, 2000.

Web links and Video Lectures:

1. https://nptel.ac.in/courses/111104098
2. https://www.youtube.com/watch?v=1YkfeR05YXY
3. https://archive.nptel.ac.in/courses/111/104/111104079/
4. https://www.youtube.com/watch?v=xGkpXk-AnWU
5. https://archive.nptel.ac.in/courses/106/104/106104170/

<table>
<tr><th rowspan="2">PCC</th><th rowspan="2">CIA</th><th rowspan="2">SEA</th><th colspan="4">CIA (50)</th><th rowspan="2">SEA
Conduction: 100 M
Reduced to: 50 M</th></tr>
<tr><th></th><th>I</th><th>II</th><th>III</th></tr>
<tr><td rowspan="5">Conduction</td><td rowspan="5">50</td><td rowspan="5">50</td><td rowspan="2">Written Test</td><td>30</td><td>30</td><td>30</td><td rowspan="4">Five questions with each of 20 marks (with internal choice). Student should answer one full question from each module</td></tr>
<tr><td colspan="3">Average of three tests – 30 Marks</td></tr>
<tr><td>Assignment</td><td colspan="3">10</td></tr>
<tr><td>AAT</td><td colspan="3">10</td></tr>
<tr><td colspan="4"><b>Total – 50 marks</b></td><td><b>Total – 50 marks</b></td></tr>
</table>

i) CIA: 50%

IA Test: 3 IA tests - Each of 30 Marks	Average of 3 tests – 30 M
Assignment	10 Marks
Additional Assessment Tools (AAT) – Quiz, Presentations, Term Paper, Open ended experiments, Mini Projects, Two-minute video on latest topic, Short MOOC courses	10 Marks
Total	**50 M**

ii) SEA : 50%

Theory Exam	5 questions to answer each of 20 Marks 2 questions from each module with internal choice Student should answer one full question from each module	20 M x 5 = **100 M reduced to 50 M**
	Total	**50 Marks**

BNM Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

SEMESTER–IV

Course: MICRO CONTROLLER AND EMBEDDED SYSTEM

CourseCode:22ISE142

L:T:P:J	**3:1:1:0**	**CIA:**	**50**
Credits:	**04**	**SEA:**	**50**
Hours:	**50**	**SEA Duration:**	**03Hours**

Course Learning Objectives: The students will be able to

1. Understand the fundamentals of ARM based systems, basic hardware components, selection methods and attributes of an embedded system.
2. ProgramARM7controller using the various instructions
3. Identify the applicability of the embedded system
4. Comprehend the real time operating system used for the embedded system Develop and test ProgramusingARM7TDMI/LPC2148

5.Conduct the experiments on an ARM7TDMI/LPC2148 evaluation board using evaluation version of Embedded 'C' &Keil Uvision-4 tool/compiler

Module-1	**No. of Hours**	**Blooms cognitive Levels**
Microprocessors versus Microcontrollers, ARM7 Embedded Systems: The RISC design philosophy, The CISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software. ARM Processor Fundamentals: Registers, Current Program Status Register, Pipeline, Exceptions, Interrupts, and the Vector Table, Core Extensions List of programs: 1.Write a program to add an array of 16 bit numbers & store the 32 bit result in internal RAM 2. Write a program to find the square of a number(1to10) using look-uptable.	**10**	**Apply**
Module-2		
Introduction to the ARM7 Instruction Set: Data Processing Instructions, Programme Instructions, Software Interrupt Instructions, Program Status Register Instruction, Coprocessor Instructions, Loading Constants ARM programming using Assembly language: Writing Assembly code, Profiling and cycle counting, Instruction scheduling, Register Allocation, Conditional Execution, Looping Constructs List of programs: 4. Write a program to find the largest number in an arrayof32 numbers. 5. Write a program to arrange a series of32bit numbers in ascending order. 6. Write a program to count the number of ones and zeros in two consecutive memory locations.	**10**	**Apply**

Module-3		
Embedded System Components: Embedded Vs General computing system, History of embedded systems, Classification of Embedded systems, Major applications areas of embedded systems, Core of an Embedded System including all types of processor/controller, Memory, Sensors, Actuators, LED,7segment LED display, stepper motor, Keyboard, Push button switch, Communication Interface (on board and external types), Embedded firmware, Other system components. List of programs: 7.Display "HelloWorld" message using Internal UART. 8.Interface and Control a DC Motor.	**10**	**Apply**
Module-4		
Embedded System Design Concepts: Characteristics and Quality Attributes of Embedded Systems, Operational quality attributes, non-operational quality attributes, Embedded Systems-Application and Domain specific, Hardware Software Co-Design and Program Modelling, embedded firmware design and development List of programs: 8.Interface a Stepper motor and rotate it in clock wise and anti-clock wise direction. 9.Interface a DAC and generate Triangular wave forms. Interface a DAC and generate Square wave forms.	**10**	**Apply**
Module-5		
RTOS and IDE for Embedded System Design: How to choose an RTOS, Integration and testing of Embedded hard ware and firmware, Embedded system Development Environment–Block diagram (excluding Keil), Disassembler / decompiler, simulator, emulator and debugging techniques, target hardware debugging, boundary scan. List of programs: 10. Demonstrate the use of an external interrupt to toggle an LED On/Off. 11. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.	**10**	**Apply**

Course Outcomes: After completing the course, the students will be able to	
CO 1	Describe the architectural features and instructions of ARM7 microcontroller
CO 2	Apply the knowledge gained for Programming ARM7 for different applications and Interface External devices and I/O with ARM7 microcontroller.
CO 3	Interpret the basic hardware components and their selection method based on the Characteristics and attributes of an embedded system.
CO 4	Develop the hardware/software co-design and firmware design approaches.
CO 5	Demonstrate the need of real time operating system for embedded system applications
CO 6	Develop and test program using ARM7 TDMI/LPC2148

Reference Books:

1. Andrew N Sloss, Dominic Symes and Chris Wright, A R M system developers guide, Elsevier, Morgan Kaufman publishers, 2008 1st Edition.
2. Shibu K V, “Introduction to Embedded Systems”, Tata Mc Graw Hill Education, Private Limited, 2nd Edition.
3. Raghunandan..G.H, Microcontroller (ARM) and Embedded System, Cengage learning Publication, 20191st edition.
4. The Insider‟s Guide to the ARM7 Based Microcontrollers, Hitex Ltd.,1stedition,2005.
5. Steve Furber, ARM System-on-Chip Architecture, Second Edition, Pearson,20151stedition.
6. Raj Kamal, Embedded System, Tata McGraw-Hill Publishers,2nd Edition,2008

Web links and Video Lectures:

1. https://academicearth.org/
2. https://onlinecourses.nptel.ac.in/noc20_ee98/preview

PCC	CIA	SEA	CIA (50)				SEA Conduction: 100 M Reduced to: 50 M
				I	II	III	
Conduction	50	50	Written Test	30	30	30	Five questions with each of 20 marks (with internal choice). Student should answer one full question from each module
				Average of three tests – 30 Marks			
			Assignment	10			
			AAT	10			
			Total – 50 marks				**Total – 50 marks**

i) CIA: 50%

IA Test: 3 IA tests - Each of 30 Marks	Average of 3 tests – 30 M
Assignment	10 Marks
Additional Assessment Tools (AAT) – Quiz, Presentations, Term Paper, Open ended experiments, Mini Projects, Two-minute video on latest topic, Short MOOC courses	10 Marks
Total	**50 M**

ii) SEA : 50%

Theory Exam	5 questions to answer each of 20 Marks 2 questions from each module with internal choice Student should answer one full question from each module	20 M x 5 = **100 M reduced to 50 M**
	Total	**50 Marks**

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

DATABASE MANAGEMENT SYSTEM
SEMESTER – IV

Subject Code: 22ISE143	Teaching Hours/Week (L: T: P: J)	3:0:1:1	CIA Marks	50
Credits		4	SEA Marks	50
Total Number of Lecture Hours		50	Exam Hours	3

Course Learning Objectives:

This course will enable students to

- Understand fundamental concepts, terminology and application of databases.
- Design concepts and creation of relational databases using relation algebra.
- Practice SQL programming through a variety of database problems.
- Demonstrate the use of Normalization, concurrency and transactions in database.

Module	Number of Hours	Bloom's Level
Module-1		
Databases and Databases Users: Characteristics of database Approach, Advantages of using the DBMS Approach **Database System Concepts and Architecture:** Data Models-Schemas and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment **Data Modeling Using the Entity-Relationship (ER) Model:** Entity Types-Entity sets-Attributes and Keys, Relationship types – Relationship Sets – Roles and structural Constraints, Weak Entity Types. **Laboratory component:** Draw ER Diagram for the following Databases using GitMind software. ORDER Database BANK Database LIBRARY Database	**10**	**Understand**
Module-2		

Relational Data Model - Concept of relations, schema-instance distinction, keys, referential integrity and foreign keys, relational algebra operators: selection, projection, cross product, various types of joins, division, example queries, tuple relation calculus, domain relational calculus, converting the database specification in E/R notation to the relational schema
Laboratory component:
Create Schema, insert at least 5 records in each table and add appropriate constraints for the following Library Database using ORACLE or MySQL DBMS under LINUX/Windows environment
BOOK(Book_id, Title, Publisher_Name, Pub_Year)
BOOK_AUTHORS(Book_id, Author_Name)
PUBLISHER(Name, Address, Phone)
BOOK_COPIES(Book_id, Branch_id, No-of_Copies)
BOOK_LENDING(Book_id, Branch_id, Card_No, Date_Out, Due_Date)
LIBRARY_BRANCH(Branch_id, Branch_Name, Address)
Write SQL queries to
1. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each branch, etc.
2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017.
	10	**Apply**

3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation. 4. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query. Create a view of all books and its number of copies that are currently available in the Library		
Module-3		
Basic SQL: SQL Data Definition and Data Types, Specifying Constraints in SQL, Basic Retrieval Queries in SQL, INSERT – DELETE and UPDATE Statements in SQL, Additional features in SQL **More SQL: Complex Queries, Triggers, Views and Schema Modification:** More Complex SQL Retrieval Queries, Specifying Constraints as Assertions and actions as Triggers, Views (Virtual Tables) in SQL, Schema Change Statements in SQL **Laboratory component:** Create Schema, insert at least 5 records for each table and add appropriate constraints for the following Order Database using ORACLE or MySQL DBMS under LINUX/Windows environment. SALESMAN(Salesman_id, Name, City, Commission) CUSTOMER(Customer_id, Cust_Name, City, Grade, Salesman_id) ORDERS(Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id) Write SQL queries to 1. Count the customers with grades above Bangalore's average. 2. Find the name and numbers of all salesman who had more than one customer. 3. List all the salesman and indicate those who have and don't havecustomers in their cities (Use UNION operation.) 4. Create a view that finds the salesman who has the customer with the highest order of a day. Demonstrate the DELETE operation by removing salesman with id1000. All his orders must also be deleted.	**10**	**Apply**
Module-4		
Basics of Functional Dependencies and Normalization for Relational Database: Functional Dependencies, Armstrong's axioms for FD's, Equivalent Decompositions, closure of a set of FDs, minimal covers, Normal forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Forms **Laboratory component:** Create Schema, insert at least 5 records for each table and add appropriate constraints for the following Company Database using ORACLE or MySQL DBMS under LINUX/Windows environment. EMPLOYEE (SSN, Name, Address, Sex, Salary, Super SSN, D No) DEPARTMENT (D No, D Name, Mgr. SSN, Mgr. Start Date) DLOCATION(D No,D Loc) PROJECT (P No, P Name, P Location, D No) WORKS_ON(SSN, P No, Hours) Write SQL queries to 1. Make a list of all project numbers for projects that involve an employee whose	**10**	**Analyze**

last name is ‘Scott’, either as a worker or as a manager of the department that controls the project. 2. Show the resulting salaries if every employee working on the ‘IoT’ project is given a 10 percent raise. 3. Find the sum of the salaries of all employees of the ‘Accounts’ department, as well as the maximum salary, the minimum salary, and the average salary in this department 4. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator). For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.		
Module-5		
Introduction to Transaction Processing –Introduction to Transaction Processing, Transaction and System Concepts, Desirable Properties on Transactions **Concurrency Control Techniques:** Transactions and Schedules, Serializability and Recoverability, Precedence Graphs, Concurrency, Lock Based Protocols, 2PL, Strict 2PL Protocols, Deadlocks - Detection and Prevention **NoSQL**: SQL v/s NoSQL, The Emergence of NoSQL, BASE Properties, Data Models: Relationships, Graph Database, Schema less Database. **Laboratory component:** Implement aggregation and indexing with suitable example using MongoDB	**10**	**Analyze**

Course outcomes: The students will able to:

CO 1:	Understand the Database System Concepts along with Data Modeling Using the Entity-Relationship (ER) Model
CO 2:	Apply the concepts of relations on RDBMS, constraints, joints using relational algebra operators.
CO 3:	Apply Structured Query Language for database manipulation.
CO 4:	Analyze functional dependencies to normalize relations of relational database
CO 5:	Analyze transactions processing, schedules protocols, serializability issues, deadlocks in DBMS and concepts of NoSQL with its advantages

Text Books: 1. Ramez Elmasari, Shamkant B Navathe “Fundamentals of Database Systems”, Pearson,Seventh Edition 2017. 2. “Database System Concepts”, Silberschatz, H Korth, S Sudarshan, 6th Edition, McGraw-Hill, 2010
Reference Books: 1. Pramod J Sadalage, Martin Fowler, “NOSQL Distilled”, Pearson, 2013

PCI	**CIA**	**SEA**	**CIA (50)**			**SEA Conduction: 100 M Reduced to: 50 M**	
				I	II	**PART A**	**PART B**
Conduction	50	50	IA Test	30	30	30 Marks	70 Marks
				Average of two tests – 30 M			
			Continuous Assessment	Weekly Assessment -20 marks			
					Total – 50 Marks	**Total – 50 Marks**	

i) CIA: 50%

IA Test: 2 IA tests - each of 30 Marks	Average of 2 tests – 30 M
Practical Lab record – 10 Marks Performance – 05 Marks Viva – 05 Marks	20 Marks
	Total 50 M

ii) SEA : 50%

Question Paper:

Theory part	5 questions to answer each of 6 Marks 2 questions from each module with internal choice Student should answer one full question from each module	6 M x 5 = 30 Marks
Execution part	Conduction - 60 Marks Viva-Voce - 10 Marks	70 Marks
	Total	**100 Marks reduced to 50 Marks**

BNM Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

SEMESTER–IV

Design and Analysis of Algorithms

Subject Code: 22ISE144	L:T:P:J: 3:0:2:0	CIE Marks:	50
Credits:	4	SEE Marks:	50
Total Number of Lecture Hours	50	SEE Duration:	3 Hours

Course objectives:

- Analyze the asymptotic performance of algorithms.
- To understand the concept of designing an algorithm.
- Synthesize efficient algorithms in common engineering design situations.

Analyze the efficiency of programs based on time complexity.

Module 1	**Teaching Hours**	**Blooms Cognitive Levels**
Introduction: Notion of algorithm, Fundamentals of Algorithmic Problem Solving, Fundamentals of the Analysis of Algorithmic Efficiency: Analysis frame work, Asymptotic Notations and BasicEfficiency Classes, Mathematical Analysis of Non-recursive and Recursive Algorithms. **Practical Programs:** 1. Implement Coin Changing problem method and find the time required. 2. Write a program to find maximum profit using Knapsack technique. 3. Implement Job Sequence problem using Greedy method.Apply brute force/divide and conquer technique to recursively implement the following concepts: a. Linear Search or Binary Search. b. To find the maximum and minimum from a given list of n elements using Brute Force Method. 4. There are 5 books in the shelf, find the number of ways to select 3 books from 5 books on the shelf using the NCR with recursion. 5. Find the next three terms of the sequence 15, 23, 38, 61, … Fibonacci series of the given number using recursion.	**10**	**Analyze**
Module 2		
Brute Force: Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching **Divide and Conquer**: General method, Binary search, Recurrence equation for divide and conquer, Finding the Maximum and Minimum, Merge sort, Quick sort, Strassen's matrix multiplication. Decrease and Conquer Approach: Topological Sort. **Practical Programs:** 1. Implement the Selection sort algorithm. 2. Implement Bubble sort algorithm. 3. Implement the Sequential Search algorithm. 4. Write a program to search a key in a given set of elements using Binary search method and find the time required to find the key. 5. Write a program to find Maximum and Minimum using divide and conquer technique and find the time required to find the elements. 6. Sort a given set of elements using Merge Sort method and determine the time required sort the elements. Plot a graph of number of elements versus time taken. Specify the time efficiency class of this algorithm. 7. Sort a given set of elements using Quick Sort method and determine the time	**10**	**Apply**

required sort the elements. Plot a graph of number of elements versus time taken. Specify the time efficiency class of this algorithm. 8. Implement Topological sort using source removal method find the time required to sort the elements.		
Module 3		
Greedy Method: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines. Minimum cost spanning trees: Prim's Algorithm, Kruskal's Algorithm. Single source shortestpaths: Dijkstra's Algorithm. Optimal Tree problem: Huffman Trees and Codes. **Practical Programs:** 1. Implement Coin Changing problem method and find the time required. 2. Write a program to find maximum profit using Knapsack technique. 3. Implement Job Sequence problem using Greedy method. 4. Implement Prim's algorithm and Find Minimum Cost Spanning Tree of a given connected undirected graph. 5. Implement Kruskal's algorithm and Find Minimum Cost Spanning Tree of a given connected undirected graph. 6. Implement Dijkstra's algorithm find shortest paths to other vertices from a given vertex in a weighted connected graph.	**10**	**Apply**
Module 4		
Dynamic Programming: General method with Examples, Multistage Graphs. Transitive Closure: Warshall's Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem, Bellman-Ford Algorithm, Travelling Sales Person problem, Reliability design. **Practical Programs:** 1. Implement all-pairs shortest paths problem using Floyd's algorithm. 2. Implement all-pairs shortest paths problem using Warshal's algorithm. 3. Implement 0/1 Knapsack using Dynamic Programming. 4. Implementation of Bellman Ford Algorithm using a directed graph. 5. Implement Travelling Sales man problem using Dynamic Programming.	**10**	**Apply**
Module 5		
Backtracking: General method , N-Queens problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles. Programme and Bound: Assignment Problem, Travelling Sales Person problem, 0/1 Knapsack problem: LC Programme and Bound solution, FIFO Programme and Bound solution. NP- Complete and NP-Hard problems: Basic concepts, nondeterministic algorithms, P, NP, NP-Complete and NP-Hard classes **Practical Programs:** 1. Implementation of N Queen Problem using Backtracking technique. 2. Implementation of SUM-SUBSET Problem. 3. Design and implement to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices using the backtracking principle. 4. Implementation Assignment Problem using Backtracking technique. 5. Implementation of Travelling Sales Man Problem using Branch and Bound method. 6. Implementation of Knapsack problem using Branch and Bound method.	**10**	**Analyze**

Course outcomes: The students should be able to:	
CO1	Apply and Analyze the asymptotic runtime complexity of algorithms by using mathematical relations that helps to identify them in specific instances
CO2	Apply and solve problems using brute force, divide and conquer techniques
CO3	Apply various problem solving methodologies such as greedy, decrease and conquer to solve a given problem.
CO4	Apply the dynamic programming to estimate the computational complexity of different algorithms.
CO5	Apply and Analyze the efficient algorithm design approaches in a problem specific manner in terms of space and time complexity

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3 rd Edition, PHI.
2. Introduction to the design and analysis of algorithms, by Anany Levitin, 3rd Edition, Pearson Education, 2011.
3. Data Structures & Algorithms using C, R.S. Salaria, 5th Edition, Khanna Publishing.
4. Computer Algorithms, by Horowitz E., Sahani S., Rajasekharan S., 2nd Edition, UniversitiesPress, 2008

Web Links:

https://www.cs.duke.edu/courses/fall08/cps230/Book.pdf.

https://www.tutorialspoint.com/design_and_analysis_of_algorithms/design_and_analysis_of_algorithms_tutorial.pdf.

PCI	**CIA**	**SEA**	**CIA (50)**			**SEA** **Conduction: 100 M** **Reduced to: 50 M**	
				I	II	**PART A**	**PART B**
Conduction	50	50	IA Test	30	30		
				Average of two tests – 30 M		30 Marks	70 Marks
			Continuous Assessment	Weekly Assessment -20 marks			
					Total – 50 Marks		**Total – 50 Marks**

i) CIA: 50%

Test: 2 IA tests - each of 30 Marks	Average of 2 tests – 30 M
Practical Lab record – 10 Marks Performance – 05 Marks Viva – 05 Marks	Marks
	Total 50 M

ii) SEA : 50%

Question Paper:

Theory part	5 questions to answer each of 6 Marks 2 questions from each module with internal choice Student should answer one full question from each module	6 M x 5 = 30 Marks
Execution part	Conduction - 60 Marks Viva-Voce - 10 Marks	Marks
	Total	**100 Marks reduced to 50 Marks**

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Computer Science and Engineering

SEMESTER– IV

Python Programming and its Application

Pre requisite: Python Programming

Subject Code: 22ISE145	L:T:P:J: 0:0:2:2	CIE Marks:	50
Credits:	2	SEE Marks:	50
Total Number of Lecture Hours	25	SEE Duration:	3 Hours

Course objectives:

1. Take a new computational problem and develop a plan to solve it through problem understanding and decomposition.
2. Follow a design creation process that includes specifications, algorithms, and testing.
3. Code, test, and debug a program in Python, based on your design

Module – 1	**Teaching Hours**
Python Basics, Entering Expressions into the Interactive Shell, The Integer, Floating-Point, and String Data Types, String Concatenation and Replication, Storing Values in Variables, **Flow control,** Boolean Values, Comparison Operators, Boolean Operators, Mixing Boolean and Comparison Operators, Elements of Flow Control, Program Execution, Flow Control Statements, Importing Modules, Ending a Program Early with sys.exit(). **List of Programs** 1. Write a python program to check whether the given string is palindrome or not. 2. Demonstrate a python code to implement abnormal termination? 3. Demonstrate a python code to print try, except and finally block statements	**5**
Module – 2	
Functions, def Statements with Parameters, Return Values and return Statements, The None Value, Keyword Arguments and print(), Local and Global Scope, The global Statement, Exception Handling, A Short Program: Guess the Number. **List of Programs** 1. Write a python program to check whether the given string is palindrome or not. 2. Write a program to create a menu with the following options 1. TO PERFORM ADDITITON 2. TO PERFORM SUBTRACTION 3. TO PERFORM MULTIPICATION 4. TO PERFORM DIVISION Accepts, users input and perform the operation accordingly. Use functions with arguments. 3. Write a python program to open a file and check what are the access permissions acquired by that file using os module?	**5**
Module – 3	
Lists, The List Data Type, working with Lists, Augmented Assignment Operators, Methods, Example Program: Magic 8 Ball with a List, List-like Types: Strings and Tuples, References, **Dictionaries and Structuring Data,** The Dictionary Data Type, Pretty Printing, Using Data Structures to Model Real-World Things. **List of Programs** 1. Create a list and perform the following methods 1) insert() 2) remove() 3) append() 4) len() 5) pop() 6) clear() 2. Create a dictionary and apply the following methods 1) Print the dictionary items 2) access items 3) use get() 4)change values 5) use len()	**5**
Module – 4	
Classes and objects, Programmer-defined types, Attributes, Rectangles, Instances as return values, Objects are mutable, Copying, **Classes and functions,** Time, Pure functions, Modifiers, Prototyping versus planning, **Classes and methods,** Object-oriented features, Printing objects, Another example, A more complicated example,The init method, The __str__ method, Operator	**5**

overloading, Type-based dispatch, Polymorphism, Interface and implementation,**Inheritance,** Card objects, Class attributes, Comparing cards, Decks, Printing the deck, Add, remove, shuffle and sort, Inheritance, Class diagrams, Data encapsulation **List of Programs** 1. Write a python Program to call data member and function using classes and objects. 2. Write a python program, which accepts the radius of a circle from user and computes the area (use math module). 3. Write a python program to create a package (college), sub-package (alldept), modules (it,cse) and create admin and cabin function to module?	
Module – 5	
NumPy and Pandas Introduction, creating Array in NumPy, Accessing of Array Elements, NumPy Array Shape, Iterating Arrays, NumPy Built in Functions, NumPy ufuncs, Creating own ufunc, Arithmetic Functions, Rounding Decimals, Finding LCM and GCD. Introduction to Pandas, Series, Key/Value as Series, Data Frames, Loading a file into Data Frame, Viewing Data, Cleaning Data, Plotting. **List of Programs** 1. Using a numpy module create an array and check the following: 1. Type of array 2. Axes of array 3. Shape of array 4. Type of elements in array 2. Using a numpy module create array and check the following: 1. List with type float 2. 3*4 array with all zeros 3. From tuple 4. Random values 3. Write a python code to read a csv file using pandas module and print the first and last five lines of a file. 4. Using a numpy module create array and check the following: 1. Reshape 3X4 array to 2X2X3 array 2. Sequence of integers from 0 to 30 with steps of 5 3. Flatten array 4. Constant value array of complex type.	**5**
MINI PROJECT	
• **Sample Python Projects:** **Project 1: Dice Rolling Simulator** This is one of the interesting python projects and will generate a random number each dice the program runs, and the users can use the dice repeatedly for as long as he wants. When the user rolls the dice, the program will generate a random number between 1 and 6 (as on a standard dice).The number will then be displayed to the user. It will also ask users if they would like to roll the dice again. The program should also include a function that can randomly grab a number within 1 to 6 and print it. **Project 2: Fantasy Cricket game:** Create a Fantasy Cricket game in Python. The game should have all the features displayed in the mock-up screens in the scenario. To calculate the points for each player, you can use rules similar to the sample rules displayed below. **Project 3: Contact Book**: Everyone uses a contact book to save contact details, including name, address, phone number, and even email address. The main objective of this project is to generate a contact book using python where users can add a new contact, edit, or delete existing contacts and view the details of all their contacts. The application should also allow users to update contact information, delete contacts, and list saved contacts. **Project 4: Spin a Yarn:** The program first prompts the user to enter a series of inputs. These can be an adjective, a preposition, a proper noun, etc. Once all the inputs are in place, they are placed in a premade story	**25**

template using concatenation. In the end, the full story is printed out to read some misintended madness! **Project 5: Rock, Paper, Scissors** A 5-minute stint of rock, paper, scissors with the computer and designed by you. We again use the random function here. You make a move first and then the program makes one. To indicate the move, you can either use a single alphabet or input an entire string. A function will have to be set up to check the validity of the move.	

Course Outcomes:

21CSE145.1: Understand the basic concepts of python programming through the Mooc courses

21CSE145.2: Create Projects using python programming for different domains.

References

1. Al Sweigart,"Automate the Boring Stuff with Python",1stEdition, No Starch Press, 2015.(Available under CC-BY-NC-SA license at https://automatetheboringstuff.com/) (Chapters 1 to 18)
2. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd Edition, Green Tea Press, 2015. (Available under CC-BY-NC license http://greenteapress.com/thinkpython2/thinkpython2.pdf) (Chapters 13, 15, 16, 17, 18) (Download pdf/html files from the above links)
3. Programming Python, Mark Lutz, O'Reilly Media, Edition 2010.
4. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372.
5. Jake VanderPlas, "Python Data Science Handbook: Essential Tools for Working with Data",1st Edition, O'Reilly Media, 2016. ISBN-13: 978-1491912058.
6. Charles Dierbach, "Introduction to Computer Science Using Python", 1st Edition, Wiley India Pvt Ltd, 2015. ISBN-13: 978-8126556014.

Semester: III / IV		
COURSE: CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS		
Course Code: 22CIP146	**L:T:P:J: 1:0:0:0**	**CIA Marks:** 50
Credits:	**1**	**SEA Marks: 5 0**
Hours:	**15 hrs**	**SEA Duration: 2Hrs**

Course Learning Objectives: The students will be able to	
1	know the fundamental political codes, structure, procedures, powers, and duties of Indian government institutions, fundamental rights, directive principles, and the duties of citizens
2	know the Indian top civil service positions and the exams conducted by UPSC and SPSC for the same
3	Understand engineering ethics and their responsibilities; identify their individual roles and ethical responsibilities towards society.

MODULE 1: Introduction to Indian Constitution	**RBT**	**Hrs**
The Necessity of the Constitution, Introduction to Indian Constitution, The Making of the Constitution, Role of Constituent Assembly, Preamble and Salient features of the Constitution of India, Fundamental Rights and its Restriction and limitations in different complex situations, Directive Principles of State Policy, Fundamental Duties.	**1,2,3**	**3**

MODULE 2: System of Government, Central Government, State Government	**RBT**	**Hrs**
System of Government-Parliamentary System, Federal System. Central Government-Basic details, Powers and Functions of Union Executive. Parliament- LS and RS (Composition, Duration, Membership and Presiding officers of Parliament and their functions). Leaders in Parliament (Leader of the House and Leader of the Opposition). Sessions of Parliament (Summoning, Adjournment, Adjournment Sine Die, Prorogation, Dissolution). Quorum of House, Language in Parliament, Joint sitting of two Houses. State Government-Basic details, Powers and Functions of State Executive. State Legislature (Composition, Duration, Membership and Presiding officers of Parliament and their functions).	**1,2,3**	**3**

MODULE 3: Judiciary, Amendments and Emergency Provisions	**RBT**	**Hrs**
Supreme Court, High Court, Judicial Review, Judicial Activism. Methods in Constitutional Amendments (How and Why). Types of Emergencies and its Consequences, Recent Amendments to the Constitution.	**1,2,3**	**3**

MODULE 4: Elections, Constitutional and Non Constitutional Bodies	**RBT**	**Hrs**
Elections- Election Commission of India, Electoral Process. Constitutional Bodies- Election Commission, Union Public Service Commission, State Public Service Commission, Goods and Service Tax Council. Non Constitutional Bodies- Central Information Commission, State Information Commission.	**1,2,3**	**3**

MODULE 5: Professional Ethics	RBT	Hrs
Scope & Aims of Engineering & Professional Ethics, Positive and Negative Faces of Engineering Ethics, Responsibilities in Engineering, the impediments to Responsibility. Trust and Reliability in Engineering, Risks, Safety and liability in Engineering, Clash of Ethics, IPRs (Intellectual Property Rights)	**1,2,3**	**3**

Course outcome: On completion of this course, students will be able to,
CO1: Have constitutional knowledge and legal literacy.
CO2: Have knowledge on All India Services and State Civil Services.
CO3: Understand Engineering and Professional Ethics and responsibilities of Engineers.

Reference Books

Suggested Learning Resources:

1.**Title of the Book - Indian Polity**
Name of the Author - M Lakshmikanth
Name of the Publisher-Mc Graw Hill Education
Edition and Year- 2019

2.**Title of the Book - Engineering Ethics**
Name of the Authors - M. Govindarajan, S.Natarajan, V.S. Senthilkumar
Name of the Publisher- Prentice-Hall
Edition and Year-2004

3. Durga Das Basu (DD Basu): "Introduction to the Constitution on India", (Students Edition.) Prentice –Hall EEE, 19th / 20th Edn., (Latest Edition) or 2008.

4. Shubham Singles, Charles E. Haries, and Et al : "Constitution of India and Professional Ethics" byCengage Learning India Private Limited, Latest Edition – 2018.

5. M.Govindarajan, S.Natarajan, V.S.Senthilkumar, "Engineering Ethics", Prentice –Hall of IndiaPvt. Ltd. New Delhi, 2004

6. M.V.Pylee, "An Introduction to Constitution of India", Vikas Publishing, 2002.

7. Latest Publications of **NHRC - Indian Institute of Human Rights**, New Delhi.

Web Links and Video Lectures

www.unacademy.com/lesson/future-perfect-tense/YQ9NSNQZ https://successesacademy

Question paper pattern for SEA and CIA.

- The SEA question paper will be set for 50 marks and the pattern of the question paper will be objective type (MCQ).
- The CIA question paper will be set for 50 marks and the pattern of the question paper will be objective type (MCQ).

Final Marks = CIA + SEA = 50+50 = 100 Marks

Class Internal Assessment

IA1	Objective type questions 50Marks	Average of 2 IA will be taken 50Marks
IA2	Objective type questions 50Marks	
	Total CIA	**50 Marks**

Semester End Assessment

Semester end Exam	Objective type questions 50Marks	50 Marks
	Total SEA	**50 Marks**

Final Marks = CIA + SEA = 50+50 = 100 Marks

BNM Institute of Technology

Syllabus for Softskills-2

SEMESTER – IV

Subject Name	Softskills-2 (Aptitude Quantitative & Logical)	**Weekly Assessment Marks**	10
Subject Code	21SFT148	**Internal Assessment Marks**	60
Number of Contact Hours/Week	3	**Company Simulation Tests Marks**	30
Total Number of Contact Hours	36	**Credits**	1
Module 1 (Quantitative Aptitude - 1)	**Number System** - Classification of Numbers, Multiple and factors, Divisibility Rules		
	HCF & LCM, Squares and Cubes.		
	Profit & Loss - Concepts of SP, CP, Profit, Loss, Gain or Loss %.		
	Profit & Loss - Marked Price & Discount problems, Successive Discount.		
	Percentages – Percent To Decimal Or Fraction Conversion, Inverse Case – Value From Percentage, relative Percentage		
	Averages - Understanding Averages & solving problems.		
Module 2 (Quantitative Aptitude - 2)	**Ratios -** Duplicate and Triplicate Ratio, Direct and Indirect variation		
	Proportion - Direct Indirect proportion and relation.		
	Simple Interest - Simple Interest, Basic Difference b/w both the Interests		
	Compound Interest - CI with a Fractional Rate, to find Instalments.		

	Speed Time & Distance - Important formulas, Relative Speed.
	Speed Time & Distance - Understanding Units & Conversion of units
Module 3 (Quantitative Aptitude - 3)	**Time & Work -** Introduction and Concept, Important Time and Work Formula, Work Done
	Time & Work - Rate of Work, Time Taken, If a piece of work is done in x number of days
	Data Interpretation - Bar Graph, Tabular Form, Line Chart, case let Form
	Data Interpretation - Pie Chart, Radar/Web, and Missing Data Interpretation.
	Probability – Understanding concepts and important formulas.
	Probability – Understanding types of problems on probability
Module 4 (Logical - 1)	Problems on Syllogisms
	Problems on Assumptions
	Logical Puzzles - K-level thinking
	Logical Puzzles - Arithmetic Puzzles
	Stick Puzzles
	Series Completion - Basics of Next no, Missing no and Wrong no and problems on that.
	Solving various types of Letter series and understanding different types.
Module 5 (Logical - 2)	**Problem on Ages -** Understanding concepts and basic formula along with solving different types of problems.
	Problem on Ages - Tips and Tricks to Solve Problems on Ages

	Blood Relation - Generation Tree, Family Tree Problems.
	Blood Relation - Statement Based Questions, Coded Blood Relation Question.
	Coding & Decoding - Concept of EJOTY, Opposite Letter, Reversing the alphabets.
	Coding & Decoding - Jumbling of Letter, Finding Codes of Derivatives.
Module 6 **(Logical - 3)**	**Clocks** – Understanding concepts and basic formula along with solving different types of problems.
	Calendar - Understanding concepts and basic formula along with solving different types of problems.
	Image Analysis - Paper cutting & Folding, Mirror & Water Image, Cubes and Dice, Analogy, Find the odd one out, Rule Detection
	Odd Man Out - Following certain patterns and groups.
	Identifying the errors/odd one in the group.
	Seating Arrangement - Linear and Circular seating Arrangements as well as problems of sitting around Square and Rectangular.
	Distance & Direction - Distance and Displacement between any two points as well as puzzles based on that, Concept of Shadows.