K. K. Wagh Institute of Engineering Education and Research, Nasik (Autonomous w.e.f. A.Y. 2022-23)
Details of Course Structure: S. Y. B. Tech Computer Engineering

● **Summary of Credits and Total Marks for U.G.Programme:**

Semester	S.Y. B.Tech	
	Total Credits (TH+PR/OR/TU)	Total Marks
III	21	725
IV	21	725
Total	**42**	**1450**

● **Description of various Courses:**

Type of Course	Description	Type of Course	Description
ESC	Engineering Science Course - Workshop - Drawing- Fundamentals of different branches	DCC	Department Core Course
BSC	Basic Science Courses	DEC	Department Elective Course
LHSM	Liberal arts, Humanities, Social Sciences and Management courses	OEC	Open Elective Courses of other technical or emerging areas /Courses designed by Industry
PSI	Project work, Seminar, Internship, PBL	IMC	Induction and Mandatory Courses
NC/AC	Non Credit Courses	ASM	Additional Specialized / MOOCs

K.K.Wagh Institute of Engineering Education and Research, Nasik (Autonomous w.e.f. A.Y. 2022-23)

Pattern of Course Structure: 2022 Semester – III S. Y. B. Tech Computer Engineering

Course Code	Course Type	Title of Course	Teaching Scheme Hrs./week			Evaluation Scheme and Marks								Credits			
			TH	TU	PR	In Sem	End Sem	CCE	TU	TW	PR	OR	Total	TH	TU	PR /OR	Total
COM222001	DCC	Fundamentals of Data Structures	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222002	DCC	Computer Graphics	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222003	DCC	Discrete Mathematics	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222004	ESC	Digital Electronics and Logic Design	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222005	DCC	Programming Paradigms and Java Programming	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222006	LHSM	Design Thinking	1	-	-	-	-	-	-	25	-	-	25	1	-	-	1
COM222007	DCC	Data Structures Lab	-	-	4	-	-	-	-	25	50	-	75	-	-	2	2
COM222008	ESC	Digital Electronics Lab	-	-	2	-	-	-	-	25	25	-	50	-	-	1	1
COM222009	DCC	Programming Paradigms and Computer Graphics Lab	-	-	2	-	-	-	-	25	25	-	50	-	-	1	1
COM222010	PSI	Python Programming Lab	-	-	2	-	-	-	-	25	-	-	25	-	-	1	1
		Total	**16**	**-**	**10**	**100**	**300**	**100**	**-**	**125**	**100**	**-**	**725**	**16**	**-**	**5**	**21**

K.K.Wagh Institute of Engineering Education and Research, Nasik (Autonomous w.e.f. A.Y. 2022-23)

Pattern of Course Structure: 2022 Semester – IV S. Y. B. Tech Computer Engineering

Course Code	Course Type	Title of Course	Teaching Scheme Hrs./week			Assessment Scheme of Marks								Credits			
			TH	TU	PR	In Sem	End Sem	CCE	TU	TW	PR	OR	Total	TH	TU	PR /OR	Total
SMH222111	BSC	Applied Mathematics –III	3	1	-	20	60	20	25	-	-	-	125	3	1	-	4
COM222012	DCC	Advanced Data Structures	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222013	DCC	Operating Systems	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222014	DCC	Computer Architecture	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222015	LHSM	Software Engineering and Project Management	3	-	-	20	60	20	-	-	-	-	100	3	-	-	3
COM222016	ASM	Client Side Technology	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COM222017	DCC	Advanced Data Structures Lab	-	-	4	-	-	-	-	25	50	-	75	-	-	2	2
COM222018	DCC	Operating Systems Lab	-	-	2	-	-	-	-	25	25	-	50	-	-	1	1
COM222019	DCC	Microprocessors Lab	-	-	2	-	-	-	-	25	25	-	50	-	-	1	1
COM222020	PSI	Project Based Learning - Client Side Technology	-	-	2	-	-	-	-	25	-	-	25	-	-	1	1
		Total	**16**	**1**	**10**	**100**	**300**	**100**	**25**	**100**	**100**	**-**	**725**	**15**	**1**	**5**	**21**

K. K. Wagh Institute of Engineering Education and Research, Nashik

(Autonomous from Academic Year 2022-23)

S. Y. B. Tech. Computer Engineering

Pattern 2022 Semester: III

COM222001: Fundamentals of Data Structures

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Theory : 03 hrs/week	**03**	**Continuous Comprehensive Evaluation: 20 Marks** **InSem Exam: 20 Marks** **EndSem Exam: 60 Marks**

Prerequisite Courses:- FYE221010: Programming in C, FYE221011:Programming in CPP

Companion Course:- COM222007: Data Structures Laboratory

Course Objectives:

- To understand basic concepts and terminology of algorithms and data structures
- To study data structures arrays, linked lists, stack and queues
- To learn searching and sorting methods

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Describe the fundamental concepts and terminology of data structures and algorithms, including arrays, linked lists, stacks, queues and searching and sorting algorithms	2-Understand
CO2	Demonstrate the ability to choose and implement appropriate data structures such as Array, linked list, stack and queue to solve a given problem	3-Apply
CO3	Implement algorithms for array and linked list processing such as insertion, and deletion using C++	3-Apply
CO4	Use stack and / or queue to solve the given problem	3-Apply
CO5	Compare different searching and sorting algorithms based on their performance, strengths, and limitations.	3-Apply

COURSE CONTENTS

Unit I	Introduction to Algorithms and Data Structures	(06 hrs)	COs Mapped - CO1

Algorithms-Introduction, Characteristics, Analysis of algorithms

Complexity of algorithms- Space complexity, Time complexity, Big O notation

Data, Data objects, Data types, Data structure**,** Abstract Data Types (ADT), Primitive and non-primitive, linear and nonlinear, static and dynamic, persistent and ephemeral data structures

Unit II	Sequential Organization	(08 hrs)	COs Mapped - CO1,CO2, CO3

Sequential Organization- Concept, Array as an abstract data type, Memory representation and address calculation, Inserting and deleting an element, Multidimensional arrays, Ordered lists

Single Variable Polynomial-Representation, evaluation and addition

Sparse Matrix-Sparse matrix representation, addition, simple transpose, fast transpose

String- Operations using arrays. Pattern matching algorithm- Naive pattern matching, Rabin Karp algorithm

Unit III	Linked Organization	(08 hrs)	COs Mapped - CO1, CO2, CO3

Linked lists-Concept, Linked list as an Abstract data type, Comparison of sequential and linked organizations

Realization of Linked list- using arrays, using dynamic memory management, header node, advantages and disadvantages of linked list

Linked list operations-Insert a node, delete a node, traverse, copy, reverse, concatenate, delete list

Types of linked list-Linear, circular , Doubly linked list and operations, Representation of a Polynomial using linked list

Generalized Linked List (GLL)-Concept, Representation of polynomial and sets.

Unit IV	Stacks and Queues	(08 hrs)	COs Mapped - CO1, CO2, CO4

Stacks-Concept, Stack as an ADT, Representation of stacks using array and linked list, stack operations, Multi-stacks

Applications of Stack- Polish notation, expression conversion and evaluation, Processing of function calls and Returns

Recursion- Concept, Types of recursion-Direct recursion, Indirect recursion, Tail recursion, Linear recursion, Tree recursion, Comparison of recursion and iterations, Backtracking algorithmic strategy, use of stack in backtracking

Queues- Concept, Queue as ADT, Realization of queues using arrays and linked list, Circular queue, Deque, Multi-queues, Linked queue and operations.

Applications of Queue: Scheduling, Josephus problem

Self Study- Four Queens problem.

Unit V	Searching and Sorting	(06 hrs)	COs Mapped - CO1, CO5

Searching Techniques- Sequential search, Binary search, Fibonacci search.

Sorting- Internal and external sorting, Sort order, Stability, Efficiency, Number of passes

Sorting methods- Bubble sort, Insertion sort, Selection sort, Quick sort, Shell sort, Bucket sort, Radix sort, Merge sort ,Comparison of Sorting Methods.

Self Study- Jump search.

Text Books

1. Horowitz, Sahani, Dinesh Mehata, “Fundamentals of Data Structures in C++”, Galgotia Publisher, ISBN: 8175152788, 9788175152786
2. J. Tremblay, P. Soresan, “An Introduction to data Structures with applications”, TMH Publication, 2nd Edition, 1984. ISBN:0-07-462471-7

Reference Books

1. Sartaj Sahani, “Data Structures, Algorithms and Applications in C++”, Second Edition, University Press, ISBN:9788173715228
2. G A V Pai, “Data Structures and Algorithms”, McGraw-Hill Companies, ISBN:9780070667266

Guidelines for Continuous Comprehensive Evaluation of Theory Course

Sr. No.	Components for Continuous Comprehensive Evaluation	Marks Allotted
1	Quiz on Unit-1, Unit-2, Unit-3 (Quiz 10 marks on each unit and will be converted to 10 Marks)	10
2	Theory assignment on Unit- 4 and 5 (10 marks assignment on unit 4 and 5 each and that will be converted in to 10 Marks)	10
	Total	**20**

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222002: Computer Graphics

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Theory : 03 hrs/week	**03**	**Continuous Comprehensive Evaluation:** 20 Marks **InSem Exam:** 20 Marks **EndSem Exam:** 60 Marks

Prerequisite Courses: - FYE 221001:Applied Mathematics

Companion Course:- COM222009: Programming Paradigms and Computer Graphics Lab

Course Objectives:

- To acquaint the learner with the basic concepts of Computer Graphics
- To learn the various algorithms for generating and rendering graphical figures
- To get familiar with the graphical transformation

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Explain basic concepts of computer graphics to generate line, circle & polygon	2-Understand
CO2	Make use of algorithms for polygon filling and polygon clipping	3-Apply
CO3	Apply geometric transformations on 2D and 3D objects	3-Apply
CO4	Make use of color models and hidden surface removal algorithms for rendering geometrical objects	3-Apply
CO5	Develop graphical applications using Curves and Fractals	3-Apply

COURSE CONTENTS

Unit I	Scan Conversion Algorithms and Display Files	(08hrs)	CO1

Introduction: Graphics Primitives - Pixel, Resolution, Aspect Ratio, Frame Buffer, Display Devices, Applications of Computer Graphics.
Scan conversion: Line drawing algorithms: Digital Differential Analyzer (DDA), Bresenham. Circle drawing algorithms: Bresenham.
Display Files: Structure, Algorithms and Display File Interpreter. Primitive operations on display file.
Segment: Segment table, Segment creation, closing, deleting and renaming, Visibility.

Self-Study Topic: Video Display Devices

Unit II	Polygons, Windowing and Clipping	(07hrs)	CO2

Polygons: Introduction to polygon. Inside test- Even-Odd, Winding Number.
Polygon Filling: Seed fill, Scan line fill.
Windowing and clipping: Introduction to windowing, 2-D clipping: Cohen – Sutherland line Clipping algorithm, Sutherland Hodgeman Polygon clipping algorithm, Weiler Atherton Polygon Clipping algorithm.

Unit III	2D, 3D Transformations and Projections	(07hrs)	CO3

2-D transformations: Homogeneous Coordinates, Translation, scaling, rotation and shear, rotation about an arbitrary point.
3-D transformations: Translation, scaling, rotation, rotation about an arbitrary axis.
Projections : Parallel (Oblique: Cavalier, Cabinet and orthographic: isometric, diametric, trimetric) and

Perspective (Vanishing Points – 1 point, 2 point and 3 point)

Unit IV	**Colour Models and Hidden Surface Removal**	**(07hrs)**	**CO4**

Colour models: Properties of Light, CIE chromaticity Diagram, RGB, HSV, CMY.
Hidden Surface Removal: Introduction, Back face detection and removal, Algorithms: Depth buffer (z), Depth sorts (Painter), Area subdivision (Warnock)

Self-Study Topic: Color Selection and Application

Unit V	**Curves and Fractals**	**(07hrs)**	**CO5**

Curves: Introduction, Interpolation and Approximation, Blending function, B-Spline curve, Bezier curve
Fractals: Introduction, Fractal generation: Koch curve, Hilbert curve, Applications.

Text Books

1. S. Harrington, Computer Graphics A Programming Approach, 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07 – 100472 – 6.
2. D. Rogers, Procedural Elements for Computer Graphics, 2nd Edition, Tata McGraw-Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4.

Reference Books

1. D. Rogers, J. Adams, Mathematical Elements for Computer Graphics, 2nd Edition, Tata McGrawHill Publication, 2002, ISBN 0 – 07 – 048677 – 8.
2. J. Foley, V. Dam, S. Feiner, J. Hughes, Computer Graphics Principles and Practice, 2nd Edition, Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.
3. D. Hearn, M. Baker, Computer Graphics – C Version, 2nd Edition, Pearson Education, 2002, ISBN 81 – 7808 – 794 – 4.

Guidelines for Continuous Comprehensive Evaluation of Theory Course

Sr. No.	Components for Continuous Comprehensive Evaluation	Marks Allotted
1	Quiz on Unit 1, Unit-2, Unit-4, Unit 5 (Quiz 15 marks each and will be converted to 15 Marks)	15
2	Theory assignment on Unit-3 (One Assignment on Unit III of 10 marks will be converted to 5 Marks)	5
	Total	**20**

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222003: Discrete Mathematics

Teaching Scheme:	**Credit Scheme:**	**Examination Scheme:**
Theory :03 hrs/week	**03**	**Continuous Comprehensive Evaluation: 20 Marks** **InSem Exam: 20 Marks** **EndSem Exam: 60 Marks**

Prerequisite Courses:- FYE 221001:Applied Mathematics-I

Course Objectives:

- To understand the concepts of relationsand functions
- To understand the use of propositional logic and number theory
- To study concepts of graph and trees
- To study algebraic structures

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	**Bloom's Level**
CO1	Solve problems using propositional logic and number theory	3- Apply
CO2	Use relations or functions to solve problems	3- Apply
CO3	Apply graph theory to represent data and solve associated problems	3- Apply
CO4	Apply the concepts of trees to generate minimum spanning tree and prefix code	3- Apply
CO5	Use algebraic structures to solve problems	3- Apply

COURSE CONTENTS

Unit I	**Propositional Logic and Number Theory**	**(06hrs)**	**COs Mapped - CO1**

Propositional Logic: Propositional equivalences, Predicates and quantifiers, Applications of propositional logic, Mathematical induction, Recursive definition

Number Theory: Introduction, Divisibility and Modular arithmetic, Greatest common divisors, Congruence, Applications of number theory.

Unit II	**Relations and Functions**	**(08hrs)**	**COs Mapped - CO2**

Relations: Properties, n-ary relations, Represent relations, Equivalence relations, Partial orderings, partitions, Hasse diagram, lattices, Chains and anti-chains, Closures of relations, Transitive closure and Warshall's algorithm

Functions: Types of functions, properties, Pigeonhole principle

Recurrence relations, Generating functions.

Unit III	**Graph Theory**	**(08 hrs)**	**COs Mapped - CO3**

Graph terminology, Types of graphs, Representation of graphs, Graph isomorphism, Planar graphs, Path and circuit, Euler path and circuit, Hamilton path and circuit, Single source shortest path-Dijkstra's algorithm, Maximum flow labeling algorithm.

Unit IV	**Trees**	**(07 hrs)**	**COs Mapped - CO4**
Trees terminology, Properties of tree, Prefix codes and Huffman coding, Cut sets, Tree traversal, Spanning trees , Minimum spanning tree, Kruskal’s and Prim’s algorithms.			
Unit V	**Algebraic Structures and Coding Theory**	**(07hrs)**	**COs Mapped - CO5**
The structure of algebra, Algebraic systems, Semi groups, Monoids, Groups, Homomorphism and normal subgroups, Congruence relations, Rings, Integral domains and fields, Coding theory.			

Text Books

1. Kenneth H. Rosen, “Discrete Mathematics and its Applications”, Tata McGraw-Hill, ISBN 978- 0-07-288008-3
2. C. L. Liu, Elements of Discrete Mathematics, Tata McGraw-Hill, ISBN 10:0-07-066913-9
3. Bernard Kolman, Robert C. Busby and Sharon Ross, “Discrete Mathematical Structures”, Prentice-Hall of India /Pearson, ISBN: 0132078457, 9780132078450.

Reference Books

1. N. Biggs, “Discrete Mathematics”, 3rd Ed, Oxford University Press, ISBN 0 –19-850717–8
2. NarsinghDeo, “Graph with application to Engineering and Computer Science”, Prentice Hall of India, 1990, 0 – 87692 – 145 – 4.

Guidelines for Continuous Comprehensive Evaluation of Theory Course

Sr. No.	**Components for Continuous Comprehensive Evaluation**	**Marks Allotted**
1	Quiz on Unit 1, Unit-2, Unit-4, Unit 5 (Quiz 15 marks each and will be converted to 15 Marks)	15
2	Theory assignment on Unit-3 (OneAssignment on Unit III of 10 marks will be converted to 5 Marks)	5
	Total	**20**

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222004: Digital Electronics and Logic Design

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Theory :03 hrs/week	**03**	**Continuous Comprehensive Evaluation: 20Marks** **InSem Exam: 20Marks** **EndSem Exam: 60Marks**

Prerequisite Course:- FYE221007 : Fundamentals of Electronics Engineering

Companion Course:- COM222008 Digital Electronics Lab

Course Objectives:

- To study logic minimization techniques
- To develop skills for design and implementation of combinational logic circuits
- To develop skills for design and implementation of sequential logic circuits

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Solve the problem of minimization using K Map and Quine Mc-Clusky method of Boolean expression	3-Apply
CO2	Build combinational circuits using AND-OR logic	3-Apply
CO3	Build combinational circuits using SSI and MSI logic	3-Apply
CO4	Explain applications of Flip Flops, registers and shift registers	2-Understand
CO5	Develop sequential logic circuits using Flip Flops and MSI logic	3-Apply

COURSE CONTENTS

Unit I	Logic Minimization Technique	(08hrs)	COs Mapped - CO1

Signed Binary Number Representation: Signed magnitude, 1's complement, 2's complement, Binary arithmetic, Boolean expression: sum of product and product of sum form, Don't care conditions, Minimization of Boolean expression using K-map(upto 4 variables) and Quine Mc-Clusky method

Unit II	Introduction to Combinational Circuits	(06hrs)	COs Mapped - CO2

Introduction to combinational circuits, **Codes &Code converter :** BCD, Excess-3, Gray code, Half-adder, Full adder, Half subtractor, Full subtractor, Universal adder/subtractor, 4 bit binary adder (IC 7483), Look ahead carry generator, BCD adder

Unit III	Combinational Logic Design	(06hrs)	COs Mapped - CO3

Multiplexers, Cascading multiplexers, Demultiplexers, Encoder, Decoder, Implementation of Boolean expression using multiplexer, Demultiplexer, Comparators, Parity generator and Checker.
Programmable Logic Devices: ROM, PLA, PAL

Unit IV	Introduction to Sequential Circuits	(08hrs)	COs Mapped - CO4

Difference between Combinational and Sequential Circuits, **Flip-Flops:** SR, Concept of preset & clear, Clocked-SR, JK, Master slave JK flip flop, T, D, Edge triggered and level triggered flip flops, Truth tables and excitation tables

Registers, Shift registers, Bidirectional shift register, Ring counter, Twisted ring counter, Universal shift register

Unit V	Sequential Logic Design	(08hrs)	COs Mapped - CO5

Counters: Types – Synchronous and asynchronous counters
Asynchronous Counters: Modulus of the counter, Decade counter, Up, Down and Up/Down counters
Synchronous sequential circuit design, State diagram, State assignment, State table, State reduction, Design procedure, Sequence generator and detector

Text Books

1. R. P. Jain, "Modern Digital Electronics", Fourth Edition, Tata McGraw Hill, ISBN 978-0-07-06691-16
2. Moris Mano, "Digital Logic and Computer Design", Second Edition, Pearson, ISBN: 978-8177584097

Reference Books

1. John Yarbrough, "Digital Logic applications and Design", Fourth Edition, Thomson Publication , ISBN:978-8131500583
2. Malvino, D.Leach "Digital Principles and Applications", Sixth Edition, Tata McGraw-Hill, ISBN: 978-0070601758

Guidelines for Continuous Comprehensive Evaluation of Theory Course

Sr. No.	Components for Continuous Comprehensive Evaluation	Marks Allotted
1	Quiz on Unit 1, Unit2, Unit 3 (Quiz 15 marks each and will be converted to 15 Marks)	15
2	Theory assignment on Unit 4, Unit 5 (One Assignment each on Unit 4 and Unit 5 of 10 marks will be converted to 5 Marks)	5
	Total	**20**

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222005:Programming Paradigms and Java Programming

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Theory :03 hrs/week	**03**	**Continuous Comprehensive Evaluation: 20Marks** **InSem Exam: 20Marks** **EndSem Exam: 60Marks**

Prerequisite Courses:-FYE221010:Programming in C, FYE221011:Programming in CPP

Companion Course:- COM222009: Programming Paradigms and Computer Graphics Lab

Course Objectives:

- To understand principles of programming paradigms
- To learn Object Oriented Programming (OOP) principles in Java programming
- To be familiar with the basic concepts of logical and functional programming language

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Remember and describe various programming paradigms	2-Understand
CO2	Make use of appropriate data types and control structures in Java to solve a given problem	3-Apply
CO3	Apply object oriented constructs in Java	3-Apply
CO4	Make use of exception handling and multithreading in Java	3-Apply
CO5	Compare and contrast Functional and Logic programming	4-Analyze

COURSE CONTENTS

Unit I	Introduction to Programming Paradigms	(05hrs)	COs Mapped - CO1

Language standardization: Proprietary and consensus,
Programming paradigms- Procedural, Object oriented, Functional, Logic
Properties of data types, objects, Scalar data types, Composite data types, Programming language syntax,
Stages in translation: analysis of the source program, synthesis of the object program.

Unit II	Introduction to Java programming	(07hrs)	COs Mapped - CO2

History and features of Java, Java Virtual Machine
Data Types: Signed vs. unsigned, User defined vs. primitive Data types, pointers
Arrays: One dimensional array, Multi-dimensional array, Alternative array declaration statements
Decision Making: if, else if, nested if, switch, Nested control structures: Syntax, semantics, pitfalls
Iterative Control Structures: while, do-while, for, the 'for- each': Syntax, semantics, pitfalls
Jump Statements :break and continue
String Handling: String classes and methods.
Comparison of Java and C++

Unit III	Object Oriented Programming in Java	(08hrs)	COs Mapped - CO3

Classes and Methods: Review of object oriented programming, objects, classes.

Assigning object reference variables, Introducing methods**,** constructors, Garbage collection, finalize() method

Inheritance: Member access and inheritance, Super class references, Using ‘super’ to call super class constructer, Creating a multilevel hierarchy, Method overriding, Dynamic method - dispatch, Using abstract classes

Packages and Interfaces: Defining a package, Finding packages, Access protection, Importing packages, Interfaces.

Comparison of Java and C++

Unit IV	Multithreading and Exception Handling using Java	(08hrs)	COs Mapped - CO4

Exception Handling: Types of Exceptions, Uncaught exceptions, Using try-catch, Multiple catch clauses, Nested try statements, Built-in exceptions, and Chained exceptions.

Multithreading in Java: Thread priorities, Synchronization, Messaging, Main thread, Creating a thread, Creating multiple threads.

Unit V	Logical and Functional Programming Languages	(08 hrs)	COs Mapped - CO5

LISP: Understanding symbol manipulation, Basic LISP functions, Definitions, predicates, Conditionals and scoping, Recursion and iteration, Properties list arrays and access functions, Using lambda definitions, Printing, Reading and atom manipulation

Prolog: Introduction, Syntax and semantics of prolog programs, Lists, Operators, Arithmetic, Using structures.

Text Books

1. T. W. Pratt, M. V. Zelkowitz, "Programming Languages Design and Implementation‖”, Fourth Edition, PHI, ISBN 81-203-2035-2
2. Herbert Schildt, "The Complete Reference Java", Ninth Edition, Tata McGraw Hill, ISBN: 978-0-07-180856-9
3.Ivan Bratko ,”Prolog programming for Artificial Intelligence”, Wesley publishers Limited, ISBN10: 0321417461 · ISBN13: 978-0321417466
4.Winston P.,Klaus B., Horn P.,”LISP”, Third Edition Pearson education, ISBN:81-7808-155-5

Reference Books

1. Carlo Ghezzi, Mehdi Jazayeri,“Programming Language Concepts”, Third Edition, Wiley Publication ISBN 978-81-265-1861-6.
2. Deugo, “Java Gems”, Cambridge University Press, ISBN 0521648246

Guidelines for Continuous Comprehensive Evaluationof Theory Course

Sr. No.	Components for Continuous Comprehensive Evaluation	Marks Allotted
1	Quiz on Unit 1, Unit-2, Unit-4, Unit 5 (Quiz 15 marks each and will be converted to 15 Marks)	15
2	Theory assignment on Unit-3 (One assignment on Unit III of 10 marks will be converted to 5 Marks)	5
	Total	**20**

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222006: Design Thinking

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Theory : 01 hr/week	**01**	**Term Work : 25 Marks**

Prerequisite Courses:- FYE 221015:Engineering Exploration

Course Objectives:

- To understand concepts of design thinking
- To understand the different phases of design thinking

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Explain stages and process of design thinking	2-Understand
CO2	Identify the methods to empathize and define the problem	2- Understand
CO3	Apply the ideation techniques for problem solving	3- Apply
CO4	Construct the prototype to evaluate a design	3- Apply
CO5	Apply testing techniques to improve the performance	3- Apply

COURSE CONTENTS

Unit I	Overview of Design Thinking Process	(02 hrs)	COs Mapped - CO1

Introduction to Design Thinking - Definition, Ideas, Inventions, Innovations, Origin of design thinking, Importance of design thinking, Problem solving, Design thinking tools.

Human-Centered Design (HCD) process - Empathize, Define, Ideate, Prototype and Test.

Unit II	Empathy and Define	(02 hrs)	COs Mapped - CO2

Empathy - How to emphasize, Role of empathy in Design Thinking, Purpose of empathy maps, Things to be done prior to empathy mapping, Customer journey mapping.

Define - How might we questions, The Five Whys Method.

Unit III	Ideation	(02 hrs)	COs Mapped - CO3

Idea generation - Basic design directions, Themes of Thinking, Inspiration and references, Brainstorming, Value, Inclusion, Sketching, Presenting ideas, Refinement, Thinking in images, Thinking in signs, Appropriation, Humour, Personification, Visual metaphors.

Unit IV	Prototype	(02 hrs)	COs Mapped - CO4

Prototyping - Assumptions during the design thinking process, Storyboards, Models and prototypes, Quick and Dirty Prototyping, Validation in the market, Best practices of presentation.

Unit V	**Testing and Implementation**	**(02 hrs)**	**COs Mapped - CO5**
Test Phase – Technique for interviews and surveys, Kano Model, Desirability testing, Testing prototypes, Obtaining feedback to refine product usability **Implementation** - Efficiency and effectiveness of innovation and implementation strategies.			

Text Books

1. Gavin Ambrose, Paul Harris “Design Thinking”, AVA Publishing (UK) Ltd, ISBN: 978-2-940411-17-7.
2. Christian Mueller Rotenberg, “Handbook of Design Thinking - Tips & Tools for how to Design Thinking”,
3. Tim Brown, Harper Collins, “Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation”, Publication, ISBN: 9780061937743.

Reference Books

1. Idris Mootee, “Design Thinking for Strategic Innovation: What They Can’t Teach You at Business or Design School”, Wiley, ISBN: 978-1-118-62012-0
2. Jeanne Liedtka and Tim Ogilvie, “Designing for Growth: A Design Thinking Tool Kit for Managers”, Columbia University Press, ISBN: 0231158386, 9780231158381

MOOC Course

1.Design Thinking - A Primer: Prof. Bala Ramadurai - https://archive.nptel.ac.in/courses/110/106/110106124

Guidelines for Term work Assessment

Term work Assessment shall be based on overall performance of a student.
Rubrics for Assessment:
R1- Multiple Choice Questions (Through ICT Tools / Paper work) (05)
R2- Case Study Presentation (10)
R3- Assignments/ Poster Presentation (10)

K.K.Wagh Institute of Engineering Education and Research, Nashik
(Autonomous from Academic Year 2022-23)

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222007: Data Structures Lab

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Practical : 04 hrs/week	02	Term Work: 25 Marks Practical Exam : 50 Marks

Prerequisite Courses:- FYE221010: Programming in C, FYE221011: Programming in CPP

Companion Course:- COM222001: Fundamentals of Data Structures

Course Objectives:

- To understand basic concepts and terminology of algorithms and data structures
- To study data structures arrays, linked lists, stack and queues
- To learn searching and sorting methods

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Describe the fundamental concepts and terminology of data structures and algorithms, including arrays, linked lists, stacks, queues and searching and sorting algorithms	2-Understand
CO2	Demonstrate the ability to choose and implement appropriate data structures such as Array, linked list, stack and queue to solve a given problem	3-Apply
CO3	Implement algorithms for array and linked list processing such as insertion, and deletion using C++	3-Apply
CO4	Use stack and / or queue to solve the given problem	3-Apply
CO5	Compare different searching and sorting algorithms based on their performance, strengths, and limitations.	3-Apply

List of Laboratory Experiments / Assignments

Sr. No.	Laboratory Experiments / Assignments	COs Mapped
1	**Set operations:** Write a menu driven C++ program to store sets for students' names participating in different events in Equinox such as Coding contest, Project competition, Paper Presentation, MasterMind etc. 1. Find out participants who have participated in Coding and Project both 2. Find out participants who have participated in Coding or Project competition or both or Mastermind 3. Find out participants who have participated in Coding but not in Master mind Find out participants who have participated in all events	CO1,CO2, CO3

2	**Knight's tour:** The problem is to move the knight, beginning from any given square on the chessboard, in such a manner that it travels successively to all 64 squares, touching each square once and only once. It is convenient to represent a solution by placing the numbers 1,2, ...,64 in the squares of the chessboard indicating the order in which the squares are reached. Note that it is not required that the knight be able to reach the initial position by one more move; if this is possible the knight's tour is called re-entrant. One of the more ingenious methods for solving the problem of the knight's tour is that given by J. C. Warnsdorff in 1823. His rule is that the knight must always be moved to one of the squares from which there are the fewest exits to squares not already traversed. Write a C++ program to implement Warnsdorff's rule and show it graphically. OR **Random walk**: A (drunken) cockroach is placed on a given square in the middle of a tile floor in a rectangular room of size n x m tiles. The bug wanders (possibly in search of an aspirin) randomly from tile to tile throughout the room. Assuming that it may move from his present tile to any of the eight tiles surrounding it (unless it is against a wall) with equal probability, how long will it take him to touch every tile on the floor at least once? Write a C++ program to graphically show a random walk of a (drunken) cockroach and find the no of moves made.	CO1,CO2, CO3
3	**String Operations:** Write a menu driven C++ program with a class for String. Write functions 1. To determine the frequency of occurrence of a particular character in the string. 2. Extract a new string from original string by accepting starting position and length 3. To accept any character and return the string with by removing all occurrences of a character accepted 4. To make an in-place replacement of a substring w of a string by the string x. Note that w may not be of same size that of x 5. To check whether given string is palindrome or not	CO1,CO2, CO3
4	**Sparse Matrix:** Write a menu driven C++ program with class for Sparse Matrix. Write functions to perform Sparse Matrix operations as listed below 1. Read sparse matrix 2. Display sparse matrix 3. Add two sparse matrices 4. Find transpose using Simple transpose algorithm 5. Find transpose using Fast transpose algorithm Compare complexity of simple and fast transpose using counter.	CO1,CO2, CO3

5	**Polynomial operations:** Write a menu driven C++ program with class for single variable polynomial and write functions to perform following polynomial operations using arrays 1. Read polynomial 2. Display polynomial 3. Add two polynomials You can try above polynomial operation using Linked list	CO1,CO2, CO3
6	**Linked list operations:** Create a linked list of names and birthdays of students. Write a menu driven C++ program to perform following operations 1. Insert name and birthday of new student 2. Delete a student entry 3. Display a happy birthday message for whom today (based on system date) is birthday 4. Display list of students with their birthdays	CO1,CO2, CO3
7	**Appointment Management:** Write a menu driven C++ program for storing appointment schedules for the day. Appointments are booked randomly using linked lists. Set start and end time for visit slots. Write functions for 1. Display free slots 2. Book appointment 3. Cancel appointment (check validity, time bounds, availability etc) 4. Sort list based on time 5. Sort list based on time using pointer manipulation	CO1,CO2, CO3
8	**Expression conversion**: Write a menu driven C++ program for expression conversion and evaluation 1. infix to prefix 2. prefix to postfix 3. prefix to infix 4. postfix to infix 5. postfix to prefix	CO1,CO2, CO4
9	**String operations:** A palindrome is a string of characters that's identical when read in forward and backward direction. Typically, punctuation, capitalization, and spaces are ignored. For example, "1.Poor Dan is in a droop!!" is a palindrome, as can be seen by examining the characters "poordanisinadroop" and observing that they are identical when read forward and backward directions. One way to check for a palindrome is to reverse the characters in the string and compare them with the original-in a palindrome, the sequence will be identical. Write C++ program with functions using Standard Template Library (STL) stack- 1. To print original string followed by reversed string using stack 2. To check whether given string is palindrome or not	CO1,CO2, CO4
10	**Simulation of pizza parlor**: Pizza parlor accepting maximum M orders. Orders are served on a first come first served basis. Order once placed cannot be canceled. Write C++ program to simulate the system using simple queue or circular queue	CO1,CO2, CO4

11	**Sorting:** Write a C++ menu driven program to store the percentage of marks obtained by the students in an array. Write function for sorting array of floating point numbers in ascending order using 1. Selection Sort 2. Bubble sort 3. Insertion sort 4. Shell Sort 5. Quick sort 6. Radix sort 7. Display top five scores Implement 4 methods of sorting. Provide choice to user to take input from user or using random numbers. Use Standard Template Library (STL) sort function for above data.	CO1, CO5
12	**Searching:** Write a C++ program to store roll numbers of students in an array who attended online lectures in random order. Write function for searching, whether a particular student attended lecture or not using 1. Linear search 2. Binary search 3. Jump search compare the searching methods based on complexities of an algorithm Provide choice to user to take input from user or using random numbers Use Visual C++ compiler to compile and execute the program.	CO1, CO5
13	A list of data representing various environmental parameters such as temperature, humidity, pollution levels, etc is maintained using appropriate data structure. Write a C++ program that uses data structures to perform the following operations: 1. Find the maximum and minimum values of each parameter in the list. 2. Calculate the average value of each parameter in the list. 3. Sort the list in ascending order of any one parameter. 4. Find the highest and lowest values of any one parameter that are considered safe for the environment. 5. Calculate the impact of the parameter values on the environment based on certain pre-defined criteria. 6. Analyze the impact of the environmental parameters on the health and safety of the society. 7. Ensure that the program follows ethical and professional practices, such as ensuring the privacy and security of the data. You should implement the program using appropriate data structures that take into account the size and complexity of the data, and demonstrate an understanding of the societal and environmental issues related to the data. Your program should also demonstrate an understanding of the impact of the parameter values on the environment, and the need for sustainable development. Finally, your program should adhere to ethical principles and professional practices, such as ensuring the confidentiality, privacy, and security of the data	CO1, CO5

Mini Project		
	Develop a mini project in a group Following is the sample problem statements based on concepts learned in the course 1. Implement an efficient system to monitor and analyze sound pollution levels in a given area. The system should be able to store and process large amounts of sound data, and provide relevant insights and visualizations to help identify areas of high sound pollution. The system should have the following functionalities: • Data Collection: Collected sound data from various sources, such as sound sensors or microphones is stored in a structured format as a file system. • Data Processing: The system should be able to process the collected data to identify patterns and trends in sound pollution levels. This could involve tasks such as noise filtering, signal processing, and feature extraction. • Data Analysis: The system should be able to analyze the processed data to provide insights into sound pollution levels in a given area. This could involve tasks such as trend analysis, outlier detection, and clustering. • Visualization: The system should be able to provide relevant visualizations to help identify areas of high sound pollution. This could involve tasks such as heat map generation, time-series plotting, and spatial analysis. The system should be designed to handle large volumes of sound data efficiently and provide real-time or near-real-time analysis and visualization. The implementation of the system should be efficient in terms of space and time complexity, and should be scalable to handle increasing volumes of data. Students are free to implement any other relevant mini project problem statement as follows. 2. Operations on Big number 3. Appointment management 4. Phone book operations 5. Sorting methods simulation and comparison	CO1 to CO5
Additional programming problems for practice		
1	**Binary Number operations:** Write a C++menu driven program for storing binary numbers using doubly linked lists. Write functions- 1. To compute 1‘s and 2‘s complement 2. Add two binary numbers	CO1, CO2, CO3
2	**GLL:** Write C++ program to realize set using generalized linked list e.g. A ={ a, b, {c, d, e, {}, {f, g}, h, i, {j, k}, l, m}. Store and print as set notation.	CO1, CO2, CO3

3	**Eight Queens:** A classic problem that can be solved by backtracking is called the Eight Queens problem, which comes from the game of chess. The chess board consists of 64 squares arranged in an 8 by 8 grid. The board normally alternates between black and white squares, but this is not relevant for the present problem. The queen can move as far as she wants in any direction, as long as she follows a straight line, Vertically, horizontally, or diagonally. Write C++ program with a recursive function for generating all possible configurations for 8-queen's problem.	CO1, CO2, CO4
4	**DEQUE:** A double-ended queue (deque) is a linear list in which additions and deletions may be made at either end. Obtain a data representation mapping a deque into a one-dimensional array. Write C++ menu driven program to simulate deque with functions to add and delete elements from either end of the deque. Also implement using STL	CO1, CO2, CO4

Guidelines for Laboratory Conduction

Use of coding standards and Hungarian notation, proper indentation and comments.
Use of open source software is to be encouraged.
Operating System recommended:- Linux or its derivative
Programming tools recommended: - Open Source line gcc/g++ (Visual C++ compiler for few assignments and note the difference)

Guidelines for Student's Lab Journal

The laboratory assignments are to be submitted by students in the form of a journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, problem statement, theory concepts in brief, algorithm, flowchart, test cases and conclusions). Program codes with sample outputs shall be submitted in soft form

Guidelines for Termwork Assessment

Continuous assessment of laboratory work shall be based on the overall performance of a student.
Assessment of each laboratory assignment shall be based on rubrics that include
R1- timely completion (10),
R2- understanding of assignment (10) and
R3- presentation/clarity of journal writing (10) (Coding standard, Indentation, Hungarian notation, input validation etc)
Mini Project assessment will be based on Teamwork, Communication skill, Social relevance of mini project, Ethics followed.

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222008: Digital Electronics Lab

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Practical : 02 hrs/week	01	Term Work: 25 Marks Practical Exam: 25 Marks

Prerequisite Courses:- FYE221007: Fundamentals of Electronics Engineering

Companion Course:- COM222004 Digital Electronics Design

Course Objectives:

- To study logic minimization techniques
- To develop skills for design and implementation of combinational logic circuits
- To develop skills for design and implementation of sequential logic circuits

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom’s Level
CO1	Solve the problem of minimization using K Map and Quine Mc-Clusky method of Boolean expression	3-Apply
CO2	Build combinational circuits using AND-OR logic	3-Apply
CO3	Build combinational circuits using SSI and MSI logic	3-Apply
CO4	Explain applications of Flip Flops, registers and shift registers	2-Understand
CO5	Develop sequential logic circuits using Flip Flops and MSI logic	3-Apply

List of Laboratory Experiments / Assignments

Sr. No.	Laboratory Experiments / Assignments	COs Mapped
1.	To Realize Full Adder and Subtractor using logic gates	CO1,CO2
2.	Design and implement Code Converters-Binary to Gray and BCD to Excess-3	CO1,CO2
3.	Design and implement of BCD Adder using 4-bit Binary Adder (IC 7483)	CO1,CO2,CO3
4.	Realization of Boolean Expression using Multiplexer	CO3
5.	Design and implement two bit comparator using logic gates	CO1, CO2
6.	Design and implement Parity Generator and checker	CO1, CO2
7.	Realization of Boolean Expression using Encoder	CO3
8.	Realization of Boolean Expression using Decoder	CO3
9.	Implement 2 bit Ripple Counter using JK Flip Flop	CO4, CO5
10.	Design of Synchronous 2 bit Up/Down Counter using JK Flip Flop	CO1, CO4, CO5
11.	Design and implement Modulo-N counter using Decade Counter IC 7490	CO1, CO4,CO5
12.	Design and implement Sequence generator and detector using JK Flip Flop	CO1, CO4, CO5
13.	Implement 3/4 bits shift registers using D Flip Flop	CO4

Guidelines for Laboratory Conduction

1. Teacher will brief the given experiment to students its procedure
2. Apparatus and equipments required for the allotted experiment will be provided by the lab assistants using SOP
3. Students will perform the allotted experiment in a group (three/four students in each group) under the supervision of faculty and lab assistant
4. After performing the experiment students will check their output from the teacher

Guidelines for Student's Lab Journal

Write-up should include title, aim, steps of circuit designing (Block Diagram, Truth Table, K Map, Expression, Realization, Conclusion)

Guidelines for Termwork Assessment

Continuous assessment of laboratory work shall be based on the overall performance of a student. Assessment of each laboratory assignment shall be based on rubrics that include

R1- timely completion (10),

R2- understanding of assignment (10) and

R3- presentation/clarity of journal writing (10)

K.K.Wagh Institute of Engineering Education and Research, Nashik
(Autonomous from Academic Year 2022-23)

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222009: Programming Paradigms and Computer Graphics Lab

Teaching Scheme:	**Credit Scheme:**	**Examination Scheme:**
Practical : 02 hrs/week	**01**	**Term Work:** 25Marks **Practical Exam:** 25Marks

Prerequisite Courses:- FYE221010:Programming in C, FYE221011: Programming in CPP, FYE221001: Applied Mathematics I

Companion Course:- COM222005:Programming Paradigms and Java Programming, COM222002:Computer Graphics

Course Objectives:

- To understand object-oriented concepts in Java such as data abstraction, encapsulation, inheritance, dynamic binding, and polymorphism
- To be familiar with functional and logical programming paradigm
- To understand basic concepts of graphics Programming
- To know various algorithms for generating and rendering geometrical objects

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	**Bloom's Level**
CO1	Demonstrate Object Oriented Programming features like inheritance, data abstraction, encapsulation, and polymorphism to solve various computing problems	2 - Understand
CO2	Illustrate the use of exception handling and multithreading in Java	2 - Understand
CO3	Compare and contrast Functional and Logic programming	2 - Understand
CO4	Apply basic concepts of computer graphics to generate line, circle and polygon	3-Apply
CO5	Make use of algorithms for polygon filling and clipping	3-Apply
CO6	Apply geometric transformations on 2D objects	3-Apply
CO7	Develop graphical applications using Curves and Fractals	3-Apply

List of Laboratory Experiments / Assignments

Sr. No.	**Laboratory Experiments / Assignments**	**COs Mapped**
1	Write a JAVA program to create a base class "Person" with name and phone number as its attributes. Derive a class "Academic Performance" with Degree and percentage as its attributes from the "Person" class. Display both personal and academic information. Make use of constructor, default constructor, copy constructor and a destructor. **Also Derive a class "Sports performance" with sports-name and score as its attribute from the "Person" class. Display personal data along with information about scores obtained in the Sport event.**	**CO1**
2	A publishing company deals with marketing of books and audio cassettes. For each book and the audio cassette the company needs to record a title and price of publication. Also a page count should be recorded for each book and a play-time in minutes should be recorded for each cassette. Design a suitable class hierarchy. Write a menu driven program that instantiates the book and tape class, allows users to manipulate and display	**CO1**

	the information about books and cassettes. The program should catch exceptions and if an exception is caught, it should replace all the values of data members with zeroes.	
3	Write a JAVA program to create User defined exception to check the following conditions and throw the exception if the criterion does not met. a. User has age between 18 and 55 b. User stays has income between Rs. 50,000 – Rs. 1,00,000 per month c. User stays in Pune / Mumbai/ Bangalore / Chennai d. User has 4-wheeler Accept age, Income, City, Vehicle from the user and check for the conditions mentioned above. If any of the condition not met then throw the exception.	**CO2**
4	Write java program to create a super class called Figure that receives the dimensions of two dimensional objects. It also defines a method called area that computes the area of an object. The program derives two subclasses from Figure. The first is Rectangle and second is Triangle. Each of the sub class overridden area() so that it returns the area of a rectangle and a triangle respectively	**CO1**
5	Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication	**CO2**
6	Consider a database of facts that describe parent relationships as well as gender relationships. The predicate parent (john,ann) is interpreted as: "John is a parent of Ann". The predicate male (john) is interpreted as: "John is a man". The predicate female (ann) is interpreted as: "Ann is a woman". Write a Prolog predicate halfsister (X,Y) that is true if X is Y's half-sister.	**CO3**
7	Declare a global constant PI and later using this value calculates the area of a circle using LISP.	**CO3**
8	Write a C++ program to draw the given pattern. Use DDA Line and Bresenham's Circle drawing algorithm.	**CO4**
9	Write a C++ program to draw a polygon and fill it with desired color using scan fill algorithm.	**CO5**
10	Write a C++ program to implement Cohen-Sutherland line clipping algorithm for a given window.	**CO5**
11	Write a menu driven C++ program to draw 2-D object and perform following transformations, a) Scaling b) Translation c) Rotation.	**CO6**
12	Write a C++ program to generate fractal patterns using Koch Curve	**CO7**

Guidelines for Laboratory Conduction

Use of open source software is encouraged.
Operating System recommended :- 64-bit Open source Linux or its derivative
Programming tools recommended: - Open Source Java Programming tool
Students shall use popular Java compilers/IDE such as GNU/Javac/Eclipse/Rose/SmartEiffel

Guidelines for Student's Lab Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of certificate, table of contents, and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Software and Hardware requirements, Theory- Concept in brief, algorithm, flowchart, mathematical model (if applicable) and conclusions. Program codes with sample output of all performed assignments are to be submitted as softcopy.

Guidelines for Termwork Assessment

Continuous assessment of laboratory work shall be based on the overall performance of a student.

Assessment of each laboratory assignment shall be based on rubrics that include

R1- timely completion (10),

R2- understanding of assignment (10) and

R3- presentation/clarity of journal writing (10) (Coding standard, Indentation, Hungarian notation, input validation etc)

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: III
COM222010: Python Programming Lab

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Practical : 2 hrs/week	**01**	**Term Work : 25 Marks**

Prerequisite Courses:- FYE221010: Programming in C, FYE221011:Programming in CPP

Course Objectives:

- To understand core python programming
- To understand python looping, control statements and string manipulations
- To understand the basic concepts of functions

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Use the core concepts to write a python program	3-Apply
CO2	Apply control structure and loops to build a solution for a given problem	3-Apply
CO3	Develop python program for string manipulation	3-Apply
CO4	Build a solution for a given problem using lists, sets, tuples, dictionaries	3-Apply
CO5	Develop programs using functions	3-Apply

COURSE CONTENTS

Installation of Python IDEs: PyCharm/Eclipse/PyDev

Data-types in Python

Variables in Python

Identifiers, Data Types, Constants, Input / Output, Operators (Arithmetic, relational, logical, bitwise), Expressions, Precedence and Associativity, Type conversions

Taking User Input (Console)

Conditional algorithmic constructs: if, if-else, nested if-else, cascaded if-else and switch statement

Iterative algorithmic constructs: 'for', 'while' statements, nested loops, Continue, break statements

Function: definition, call, variable scope and lifetime, the return statement. Defining functions, Lambda or anonymous function

Arrays: One- dimensional, multidimensional array, character arrays (Strings).

List of Laboratory Experiments / Assignments		
Sr. No.	**Laboratory Experiments / Assignments**	**CO Mapped**
1	Write a python program that accepts seconds as input of type integer. The program should convert seconds in hours, minutes and seconds. Output should like this : Enter seconds: 12200 Hours: 3 Minutes: 23 Seconds: 20	**CO1**
2	**Conditional Structures** The marks obtained by a student in 3 different subjects are input by the user. Python program should calculate the average marks obtained in 3 subjects and display the grade. The student gets a grade as per the following rules: Average Grade 90-100 O 80-89 A 70-79 B 60-69 C 40-59 D 0-39 F	**CO2**
3	**Control structures** Floyd's triangle is a right-angled triangular array of natural numbers as shown below: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 Write a python program to print the Floyd’s triangle	**CO2**
4	**String** Write a python program that accepts a string to setup a password with following requirements: • The password must be at least eight characters long • It must contain at least one uppercase letter • It must contain at least one lowercase letter • It must contain at least one numeric digit The program checks the validity of password.	**CO3**
5	**List** Write a python program to • Find the sum and average of given numbers using lists • Display elements of list in reverse order • Find the minimum and maximum elements in the lists	**CO4**
6	**Tuple** Write a Python program to sort a tuple by its float element. Sample data: [('item1', '13.10'), ('item2', '17.10'), ('item3', '25.3')] Expected Output: [('item3', '25.3'), ('item2', '17.10'), ('item1', '13.10')]	**CO4**

7	**Dictionary** Write a python program to read string from user and create a dictionary having key as word length and value is count of words of that length. For example, if user enters ‘I scream you scream we all scream for ice cream’ Word Word length I 1 scream 6 you 3 scream 6 we 2 all 3 scream 6 for 3 ice 3 cream 5 The content of dictionary should be {1:1, 6:3, 3:4, 2:1, 5:1}	**CO4**
8	**Set** Write a python program for operations on set	**CO4**
9	**Function** Write a function in python to display the elements of list thrice if it is a number and display the element terminated with ‘#’ if it is not a number. Suppose the following input is supplied to the program: [‘23’,‘MAN’,‘GIRIRAJ’, ‘24’,‘ZARA’] The output should be 232323 MAN# GIRIRAJ# 242424 ZARA#	**CO5**
Mini Project		
10	Develop a mini project in a group based on Python programming concepts and design thinking	**CO1 to CO5**

Guidelines for Laboratory Conduction

Use of coding standards and Hungarian notation, proper indentation and comments.
Operating System recommended:- Linux or its derivative
Use the concepts of design thinking in mini project.

Guidelines for Student's Lab Journal

The laboratory assignments are to be submitted by students in the form of a journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, problem statement, theory concepts in brief, algorithm, flowchart, test cases and conclusions). Program codes with sample outputs shall be submitted in soft form.

Guidelines for Termwork Assessment

Continuous assessment of laboratory work shall be based on the overall performance of a student. Assessment of each laboratory assignment shall be based on rubrics that include

R1- timely completion (10)
R2- understanding of assignment (10)
R3- Use Coding standards, proper documentation, neatness of writeup (10) – 5 marks for coding standards and documentation and 5 marks for neatness of write up

Text Books
1. Reema Thareja, “Python Programming Using Problem Solving Approach”, Oxford University Press, ISBN 13: 978-0-19-948017-6 2. 2. R. Nageswara Rao, “Core Python Programming”, Dreamtech Press, ISBN-13: 978-9386052308
Reference Books
1. R. G. Dromey, “How to Solve it by Computer”, Pearson Education India, ISBN-13: 978-8131705629 2. Maureen Spankle, “Problem Solving and Programming Concepts”, Pearson, ISBN-13: 978-0132492645

K. K. Wagh Institute of Engineering Education and Research, Nashik
(Autonomous fromAcademicYear2022-23)

S.Y.B.Tech. Computer Engineering
Pattern 2022 Semester: IV
SMH222111: Applied Mathematics-III

Teaching Scheme:	**Credit Scheme:**	**Examination Scheme:**
Theory: 03hrs/week **Tutorial:01hr/week**	**03** **01**	**ContinuousComprehensive Evaluation: 20Marks** **InSem Exam: 20Marks** **EndSemExam: 60Marks** **Tutorial: 25Marks**

Prerequisite Courses:- Applied Mathematics-I

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	**Bloom's Level**
CO1	Understand basic concept of Statistic	2-Understand
CO2	Understand basic concept of probability distribution	2-Understand
CO3	Apply the basic concepts of statistics to real life problems	3-Apply
CO4	Apply the basic concepts of probability distribution theory to real life problems	3-Apply
CO5	Analyze real life problems by using theory of statistics and Probability distribution	4-Analyze

COURSE CONTENTS

UnitI	**Descriptive Measures**	**(08hrs+2hrsTutorial)**	**COs Mapped - CO1, CO2, CO3**

Measures of central tendency (Mean, Median, Mode), Measures of dispersion (Variance, Standard Deviation, Range), coefficients of variation, Moments, Skewness and Kurtosis.

Unit II	**Random Variable & Distribution Functions**	**(08hrs+2hrsTutorial)**	**COs Mapped -CO1, CO2, CO3**

Random Variable, Distribution functions (Continuous and discrete), Properties of distribution function, Probability mass function (p.m.f.), Probability density function (p.d.f.) and Cumulative distribution function (Continuous and discrete).

Unit III	**Mathematical Expectation and Generating Function**	**(08hrs+2hrsTutorial)**	**COs Mapped - CO3, CO4, CO5**

Mathematical Expectation, Properties of expectation, Moment Generating Function

Unit IV	**Probability Distributions**	**(08hrs+2hrsTutorial)**	**COsMapped - CO4, CO5**

Discrete distributions: Geometric, Binomial, Poisson, Uniform Distribution Continuous distribution: Normal distribution, Standard Normal, Uniform.			
Unit V	**Correlation and Regression**	**(08hrs+2hrsTutorial)**	**COs Mapped - CO1, CO2**
Covariance, Concept of correlation, Karl Pearson’s Coefficient of Correlation, Rank Correlation coefficient, Spearman’s rank Correlation coefficient. Regression: Lines of Regression, Regression coefficients.			
TextBooks			
1. B.V.Ramana,“Higher Engineering Mathematics”, TataMcGraw-Hill. 2. B.S.Grewal,”Higher Engineering Mathematics”,Khanna Publication,Delhi. 3. AdvancedEngineeringMathematics,7e,bypeterV.O‟Neil(ThomsonLearning) 4. IntroductiontoProbabilityandStatistics forEngineersandScientists,5e, bySheldonM.Ross(ElsevierAcademicPress)			
ReferenceBooks			
1. Erwin Kreyszig,”Advanced Engineering Mathematics”,WileyEastern Ltd. 2. P.N.Wartikar and J.N.Wartikar,”Applied Mathematics”(VolumesI and II), Pune Vidyarthi Griha Prakashan, Pune. 3. AdvancedEngineeringMathematics,2e,by M.D.Greenberg(PearsonEducation).			

Guidelines for Continuous Comprehensive Evaluation of Theory Course		
Sr.No.	**Components for Continuous Comprehensive Evaluation**	**Marks Alloted**
1	Assignments (Total3Assignment, Unit I and II 20marks,Unit III and IV20marks and UnitV- 10marks &50markswillbeconvertedto10Marks)	10
2	Tests on each unit using LearniCo (Each test for15 Marks and total will be converted out of 10Marks)	10

List of Tutorial Assignments		
Sr.No.	Title	CO Mapped
1	Examples on measures of central tendency and measures of dispersion	CO1,CO2, CO3
2	Examples on Probability density function (p.d.f.) and Cumulative distribution function (Continuous and discrete).	CO1,CO2, CO3
3	Examples on Probability mass function (p.m.f.) and Probability density function (p.d.f.)	CO1,CO2
4	Examples on Cumulative distribution function (Continuous and discrete).	CO1,CO2
5	Solve problems on measures of central tendency using MATLAB	CO1,CO2, CO3,CO4
6	Solve problems on measures of dispersion using MATLAB	CO1,CO2, CO3,CO4
7	Examples on Mathematical Expectation, Properties of expectation,	CO1,CO2, CO3
8	Examples on Moment generating function	CO1,CO2, CO3

9	Examples on Geometric, Binomial, Poisson, Uniform Distribution	CO3, CO4,CO5
10	Examples on Normal, Standard Normal &Uniform distribution	CO3, CO4,CO5
11	Examples on Covariance, Karl Pearson’s Coefficient of Correlation, Rank Correlation coefficient, Spearman’s rank Correlation coefficient.	CO4,CO5
12	Examples on Lines of regression, Regression coefficients.	CO4,CO5

Guidelines for Tutorial/Termwork Assessment		
Sr.No.	**Components for Tutorial/Termwork Assessment**	**Marks Allotted**
1	Assignment on Computational Software	5
2	Tutorial (Each tutorial carries 15marks)	15
3	Attendance (Above95%:05Marks,below75%: 0Marks)	5

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: IV
COM222012: Advanced Data Structures

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Theory : 03 hrs/week	**03**	**Continuous Comprehensive Evaluation: 20 Marks** **InSem Exam: 20 Marks** **EndSem Exam: 60 Marks**

Prerequisite Courses:- COM222001: Fundamentals of Data structures, COM222003:Discrete Mathematics

Companion Course:- COM222017: Advanced Data Structures Laboratory

Course Objectives:

- To understand basic concepts of non linear data structures such as trees, graphs
- To study the concepts of hash table and files
- To learn advanced data structures such as indexing techniques and multiway search trees

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom’s Level
CO1	Make use of non-linear data structures such as graph and trees to solve a given problem	3-Apply
CO2	Use different representations of symbol table	3-Apply
CO3	Apply the hash table and it’s collision resolution methods and different file handling techniques	3-Apply
CO4	Use efficient indexing techniques and multiway search trees to store and maintain data	3-Apply
CO5	Analyze an algorithm used for solving a given problem	4-Analyze

COURSE CONTENTS

Unit I	Graphs	(08 hrs)	COs Mapped - CO1, CO5

Graph- Basic Concepts, Storage representation- Adjacency matrix, Adjacency list, Adjacency multi list
Traversals-Depth First Search (DFS) and Breadth First Search(BFS)
Spanning Tree - Connected components, Minimum spanning Tree, Greedy algorithms- Prim’s and Kruskal’s for MST
Dikjtra's Single source shortest path, Algorithm for Topological ordering
Self Study-Data structure used in Webgraph and Google map.

Unit II	Trees	(08 hrs)	COs Mapped - CO1, CO5

Trees- Basic terminology, General tree and its representation, Representation using sequential and linked organization**,** Converting tree to binary tree, Types of trees
Binary tree- Properties, ADT, Representation using sequential and linked organization, Binary tree traversals (recursive and non-recursive)- inorder, preorder, postorder, Depth first and breadth first search, Operations on binary tree, Formation of binary tree from given traversals,
Applications of Binary trees

Binary Search Tree (BST) - Concept, Definition, Comparison with binary tree, BST operations, applications of BST
Threaded binary tree, Expression tree, Huffman Tree (Concept and Use), Decision Tree, Game tree.

Unit III	Symbol Table	(07 hrs)	COs Mapped – CO2, CO5

Symbol Table-Representation of Symbol Tables- Static tree table and Dynamic tree table, Weight balanced tree - Optimal Binary Search Tree (OBST), OBST as an example of Dynamic Programming
Height Balanced Trees- AVL tree. Red-Black Tree, Splay Tree.

Unit IV	Hash tables and Files	(07 hrs)	COs Mapped – CO3, CO5

Hash table Concepts-Hash function, bucket, Collision, Probe, Synonym, Overflow, Open hashing, Closed hashing, Perfect hash function, Load density, Full table, Load factor, Rehashing, Basic operations, Issues in hashing
Hash functions- Properties of good hash function, Division, Multiplication, Extraction, Mid-square, folding and universal
Collision resolution strategies-Open addressing and Chaining, Hash table overflow- Open addressing and Chaining, Closed addressing and Separate chaining.
Files-Concept, Need, Primitive operations. Sequential file organization, Direct access file, Indexed sequential file organization-Concept and Primitive operations
Self Study- SkipList- Representation, Searching.

Unit V	Indexing and Multiway Trees	(06 hrs)	COs Mapped – CO4, CO5

Indexing and Multiway Trees- Indexing, Indexing techniques-Primary, Secondary, Dense, Sparse
Multiway search trees, B-Tree- Insertion, Deletion, B+ Tree - Insertion, Deletion, Use of B+ tree in Indexing
Heaps- Concept, Insert, Delete operation, Heap sort, Heap as a Priority Queue.
Self Study- Trie Tree

Text Books

1. Horowitz, Sahani, Dinesh Mehata, "Fundamentals of Data Structures in C++", Galgotia Publisher, ISBN: 8175152788, 9788175152786
2. M Folk, B Zoellick, G. Riccardi, "File Structures", Pearson Education, ISBN:81-7758-37-5

Reference Books

1. Sartaj Sahani, "Data Structures, Algorithms and Applications in C++", Second Edition, University Press, ISBN: 9788173715228
2. G A V Pai, "Data Structures and Algorithms", McGraw-Hill Companies, ISBN:9780070667266

Guidelines for Continuous Comprehensive Evaluation of Theory Course

Sr. No.	Components for Continuous Comprehensive Evaluation	Marks Allotted
1	Quiz on Unit 1, Unit-2, Unit-3 (Quiz 10 marks on each unit and will be converted to 10 Marks)	10
2	Theory assignment on Unit- 4 & 5 (10 marks assignment on unit 4 and 5 each and that will be converted in to 10 Marks)	10
	Total	**20**

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: IV
COM222013 : Operating systems

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Theory : 03 hrs/week	03	Continuous Comprehensive Evaluation: 20 Marks InSem Exam: 20 Marks EndSem Exam: 60 Marks

Prerequisite Courses:- COM222001:Fundamentals of data structures

Companion Course:- COM222018: Operating Systems Lab

Course Objectives:

- To understand operating system services, types of operating systems and shell scripts
- To study process scheduling algorithms and multithreading techniques
- To get acquainted with the concepts of synchronization, deadlock prevention and avoidance algorithms
- To learn concepts of memory management and I/O management techniques
- To introduce Linux operating systems

Course Outcomes: On completion of the course, students will be able to

	Course Outcomes	Bloom's Level
CO1	Explain operating system services, types of operating systems and basic shell commands	2- Understand
CO2	Illustrate the concept of process scheduling algorithms to solve scheduling problems	2- Understand
CO3	Compare algorithms for deadlock detection, prevention and avoidance	2- Understand
CO4	Use algorithms for page replacement and I/O management	3- Apply
CO5	Describe Linux commands and utilities such as grep, tr, sed, awk	2- Understand

COURSE CONTENTS

Unit I	Fundamental concepts of operating systems	(07 hrs)	COs Mapped - CO1

Introduction, Operating systems services
Types of operating systems: Batch, Time-sharing, Network, Distributed and real time.
Operating system operations: Dual mode and multimode, System calls, Types of system calls.
Bash shell scripting: Basic shell commands and scripting language.

Unit II	Process management	(08 hrs)	COs Mapped - CO2

Process: Concept, Process control block, Process state diagram, Inter process communication
Process scheduling: Types, First come first serve, Shortest job first, Round robin, Priority based scheduling
Threads: Multi core programming, Multithreading models, Implicit threading, Threading issues

Unit III	Process coordination	(07 hrs)	COs Mapped - CO3

Synchronization: The critical-section problem, Peterson's solution, Synchronization hardware, Mutex locks, Semaphores, Monitors

<table>
<tr><td colspan="4"><b>Classic problems of synchronization:</b> Producer-consumer problem, Reader/writer problem, Dining philosopher problem
<b>Deadlock:</b> Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance and detection, Recovery from deadlock.</td></tr>
<tr><td><b>Unit IV</b></td><td><b>Memory Management</b></td><td><b>(07 hrs)</b></td><td><b>COs Mapped - CO4</b></td></tr>
<tr><td colspan="4"><b>Memory Partitioning</b>: Fixed partitioning, Dynamic partitioning
<b>Contiguous Memory allocation techniques:</b> First fit, Best fit, Worst fit, Swapping, Structure of the page table, Segmentation, Demand paging
<b>Page Replacement algorithms</b>: First in first out, Optimal page replacement, Least recently used translation look aside buffer</td></tr>
<tr><td><b>Unit V</b></td><td><b>I/O management and Introduction to Linux</b></td><td><b>07 hrs</b></td><td><b>COs Mapped – CO4, CO5</b></td></tr>
<tr><td colspan="4">I/O devices, Disk scheduling algorithms: First come first serve, Shortest seek time first algorithm, SCAN, Circular-SCAN
<b>Introduction to Linux:</b> Essential features, File systems and directories, Linux shell commands such as pwd, cd, ls, cat, rm, cp, mkdir and Linux utilities such as tr, sed, grep, egrep, awk. File access rights.</td></tr>
<tr><td colspan="4"><b>Text Books</b></td></tr>
<tr><td colspan="4">1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, WILEY, ISBN:978-81-265-5427-0, 9th Edition
2. William Stallings, “Operating System: Internals and Design Principles”, Prentice Hall, ISBN 10: 0-13-380591-3, ISBN 13: 978-0-13-380591-8, 8th Edition</td></tr>
<tr><td colspan="4"><b>Reference Books</b></td></tr>
<tr><td colspan="4">1. Tom Adelstein and Bill Lubanovic, “Linux System Administration”, O’Reilly Media, ISBN 10: 0596009526, ISBN 13: 978-0596009526
2. Harvey M. Deitel, “Operating Systems”, Prentice Hall, ISBN 10: 0131828274, ISBN 13: 978-0131828278</td></tr>
</table>

<table>
<tr><td colspan="3"><b>Guidelines for Continuous Comprehensive Evaluation of Theory Course</b></td></tr>
<tr><td><b>Sr. No.</b></td><td><b>Components for Continuous Comprehensive Evaluation</b></td><td><b>Marks Allotted</b></td></tr>
<tr><td>1</td><td>Quiz on Unit 1, Unit-2, Unit-4, Unit 5
(Quiz 15 marks each and will be converted to 15 Marks)</td><td>15</td></tr>
<tr><td>2</td><td>Theory assignment on Unit-3
(One Assignment on Unit III of 10 marks will be converted to 5 Marks)</td><td>05</td></tr>
<tr><td></td><td><b>Total</b></td><td><b>20</b></td></tr>
</table>

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: IV
COM222014: Computer Architecture

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Theory :03 hrs/week	**03**	**Continuous Comprehensive Evaluation: 20Marks** **InSem Exam: 20Marks** **EndSem Exam: 60Marks**

Prerequisite Course:- COM22204: Digital Electronics and Logic Design

Companion Course:- COM222019: Microprocessor Lab

Course Objectives:

- To study data representation, register transfer and Microoperations
- To get familiar with Micro programmed Control Unit
- To understand fundamentals of Central Processing Unit, Pipelining, Input-output and Memory Organization

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Explain data representation, register transfer and Microoperations	2-Understand
CO2	List the steps for design of accumulator logic	1-Remember
CO3	Compare hardwired and micro programmed control	2-Understand
CO4	Explain pipeline and vector processing and algorithms for arithmetic operations	2-Understand
CO5	Explain I/O and memory organization	2-Understand

COURSE CONTENTS

Unit I	Data Representation, Register Transfer and Microoperations	(08hrs)	COs Mapped - CO1

Data Representation: Fixed point and floating-point representations
Microoperations: High definition language, Register transfer language, Register transfer, Bus and memory transfers, Arithmetic micro operations, Logic micro operations, Shift micro operation

Unit II	Basic Computer Organization and Design	(06hrs)	COs Mapped - CO2

Instruction Codes, Registers, Instructions, Timing and control, Instruction cycle, Memory reference instructions, Input-Output and interrupts, Design of accumulator logic

Unit III	Micro programmed Control	(06hrs)	COs Mapped - CO3

Introduction to assembly language, Program loops, Arithmetic and logical operations, Sub-routines, Input-output, Control memory, Address sequencing, Micro program example, Design of control unit

Unit IV	CPU, Pipeline Processing	(08hrs)	COs Mapped - CO4

CPU: Register and stack organization, Instruction formats, Addressing modes, Data transfer and manipulation, Program control, RISC and CISC
Pipeline Processing: Pipelining, Arithmetic pipeline, Instruction pipeline, RISC pipeline, Vector processing

Unit V	I/O and Memory Organization	(08hrs)	COs Mapped - CO5

Input-Output: Interface, Asynchronous data transfer, Modes of transfer, Priority interrupt, Direct memory access, Input output processor, Serial communication
Memory: Hierarchy, Main memory, Auxiliary memory, Associative memory, Cache memory, Virtual memory, Memory management hardware

Text Books

1. Morris Mano, “Computer System Architecture”, PHI, Third Edition, ISBN- 81-7808-687-5
2. Zaky S, Hamacher, “Computer Organization”, McGraw-Hill Publications, Fifth Edition, 2001, ISBN- 978-1-25-900537-5

Reference Books

1. W. Stallings, “Computer Organization and Architecture: Designing for Performance”, Eighth Edition, Prentice Hall of India, ISBN 13: 978-0-13-607373-4
2. Peter Abel, Niyaz Nizamuddin, “IBM PC Assembly Language and Programming”, Fifth edition, Pearson, ISBN: 9 798177 586564

Guidelines for Continuous Comprehensive Evaluation of Theory Course

Sr. No.	Components for Continuous Comprehensive Evaluation	Marks Allotted
1	Quiz on Unit 1, Unit 2, Unit 3 (Quiz 15 marks each and will be converted to 15 Marks)	15
2	Theory assignment on Unit 4, Unit 5 (One Assignment each on Unit 4 and Unit 5 of 10 marks will be converted to 5 Marks)	5
	Total	**20**

K. Wagh Institute of Engineering Education and Research, Nashik
(Autonomous from Academic Year 2022-23)

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: IV
COM22015: Software Engineering and Project Management

Teaching Scheme:	**Credit Scheme:**	**Examination Scheme:**
Theory : 03 hrs/week	**03**	**Continuous Comprehensive Evaluation:** 20 Marks **InSem Exam:** 20 Marks **EndSem Exam:** 60 Marks

Prerequisite Courses:-COM222001:Fundamentals of Data Structures, COM222005:Programming Paradigms and Java Programming

Course Objectives:

- To understand the need for the software life cycle and its implications
- To be acquainted with methods of capturing, specifying, visualizing and analyzing software requirements
- To understand project management through the life cycle of the project and current practices in the IT industry

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	**Bloom's Level**
CO1	Identify appropriate process model for software development.	3-Apply
CO2	Model software requirements for software development.	3-Apply
CO3	Make use of emerging trends for software project management.	3-Apply
CO4	Utilize project metrics for software project estimation and process improvement	3-Apply
CO5	Analyze software risks involved in project development.	4-Analyze

COURSE CONTENTS

Unit I	**Introduction to Software Engineering and Software Process Models**	**(08hrs)**	**CO1**

Software Engineering: The Nature of Software, Defining Software, Software Engineering Process, Software Engineering Practice.

Process Models: A Generic Process Model, Process Assessment and Improvement, Prescriptive process models.

Agile Development: Agility, Agility and Cost of change, Agile process, Extreme Programming (XP), Other Agile Process Models- Scrum, Feature Driven Development (FDD)

Self-Study Topic: Use of Agile to enhance business processes by major players such as Sky, Philips and JP Morgan Chase

Unit II	**Understanding Requirements and Design Concepts**	**(07hrs)**	**CO2**

Requirement Engineering: Establishing the Groundwork**,** Eliciting Requirements, Developing the use cases, Building the Requirement model, Negotiate Requirements, Validating Requirements, and Requirement Analysis.

Design Concepts: Design within the context of Software Engineering, The Design Process, Design Concepts, and The Design Model.

Self-Study Topic: Software Requirement Specification of Library Management System

Unit III	**Emerging Trends in Software Engineering & Project Management Concepts**	**(07hrs)**	**CO3**

Emerging Trends: Technology evolution, Observing Software Engineering Trends, Identifying soft trends, Technology directions, Tools related trends. **Project Management Concepts:** The management spectrum, People, The Product, The Process, The Project, The W^5HH Principle			
Unit IV	**Project Estimation and Software Process Improvement**	**(07hrs)**	**CO4**
Project Metrics: Software Measurement, Metrics for Software Quality, Metrics for Small Organizations **Estimation for Software Projects:** Observation on Estimation, The Project Planning Process, Software Scope and Feasibility, Resources, Software Project Estimation, Decomposition Techniques, Empirical Estimation Models, Specialized Estimation Techniques **Software Process Improvement:** Introduction, Approaches to SPI, Maturity Models - Capability Maturity Model (CMM), Capability Maturity Model Integration (CMMI)			
Unit V	**Project Scheduling and Risk Management**	**(07hrs)**	**CO5**
Project Scheduling: Basic Principles, Task set for Software Project, Task Network, Scheduling **Risk Management:** Reactive versus Proactive Risk Strategies, Software Risks, Risk Identification, Risk Projection, Risk Refinement, The RMMM Plan			
Self-Study Topic: Risk management for E-commerce website			
Text Books			
1. Roger Pressman, “Software Engineering: A Practitioner’s Approach”‖, McGraw Hill, ISBN 0-07-3375 2. Ian Sommerville,”Software Engineering”, Addison and Wesely, ISBN 0-13-703515-2.			
Reference Books			
1. Rajib Mall, “Fundamentals of Software Engineering”, Prentice Hall India, ISBN-13: 978-8120348981 2. Pankaj Jalote, “An Integrated Approach to Software Engineering”, Springer, ISBN 13: 9788173192715.			

Guidelines for Continuous Comprehensive Evaluation of Theory Course		
Sr. No.	**Components for Continuous Comprehensive Evaluation**	**Marks Allotted**
1	Quiz on Unit-1, Unit-2, Unit-4 and Unit-5 (Quiz 15marks each and will be converted to 15 marks)	15
2	Theory assignment on Unit-3 (One assignment on Unit-3 of 10 marks will be converted to 5 marks)	05
	Total	**20**

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: IV
COM222016: MOOC – Client Side Technology

Teaching Scheme:	**Credit Scheme:**	**Examination Scheme:**
Theory: MOOC	**01**	

Prerequisite Courses:- COM222006:Design Thinking

Companion Course:- COM222020: Project Based Learning –Client Side Technology

Course Objectives:

- To understand the concepts of front end web technologies.
- To understand client side technologies

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	**Bloom's Level**
CO1	Build web pages using HTML	3-Apply
CO2	Apply CSS for styling web pages	3-Apply
CO3	Use of Java Script for web development	3-Apply
CO4	Use Angular for web development	3-Apply
CO5	Use front-end frameworks for web development	3- Apply

COURSE CONTENTS

Unit I	**Client side scripting - Hyper Text Markup Language**	**(02hrs)**	**COs Mapped - CO1**
The Internet, basic internet protocols, the World Wide Web, HTTP Request message, HTTP response message, web clients, web servers. HTML: Introduction, HTML elements: headings, paragraphs, line break, colors and fonts, links, frames, lists, tables, images and forms, Difference between HTML and HTML5. https://drive.google.com/drive/folders/18j8w9gcUey0EOgDEoqVpxIdtEfA69E3p			
Unit II	**Cascaded Style Sheets**	**(02hrs)**	**COs Mapped - CO2**
Introduction to Style Sheet, CSS features, CSS core syntax, Style sheets and HTML, Style rule cascading and inheritance, text properties. CSS Color, CSS Background Image, CSS Selectors, CSS BOX model, introduction to Bootstrap. https://drive.google.com/drive/folders/1oFTQKtBlnZB3dSHOiCKgQtYY4-VkXS76			
Unit III	**Client Side Technology - Java Script**	**(02hrs)**	**COs Mapped - CO3**
Java Script: Introduction to JavaScript, Document Object Modelling, Benefits of JavaScript, Fundamentals: Variables, Constants, Data Types, Objects, Functions, Conditional Statements, Loops, Switch Case. https://drive.google.com/drive/folders/1_szX6sGFwtJ14KppmPU70flrFPVczAvp			
Unit	**Client Side Technology - Angular**	**(03hrs)**	**COs Mapped -**

IV			**CO4**
Introduction to Single Page Application, Angular, Angular routing Angular directives, Angular components , One-way data binding for read-only data, Two-way data binding, Events, Format data with pipes, Shared service, Use routing to navigate among different views and their components. https://drive.google.com/drive/folders/1GkH-8FNEm1HmC7Urr5m83znkg5_bDldw			
Unit V	**Front End Technologies**	**(03hrs)**	**COs Mapped - CO5**
Introduction to React JS, React JS installation, React Component, React Lifecycle, React Events, Introduction to Node.js, Features of Node.js, Node.js Architecture, Node.js module, JSON File, Node.js Operators, Node.js functions, Node.js Objects, Node.js file system, Node.js Events, Node.js HTTP module. https://drive.google.com/drive/folders/15EkXqxAzMe8L0Gzt9Hm1ybOkez0yE5VD			
Learning Material			
1. Jeffrey C. Jackson ,” Web Technologies: A Computer Science Perspective”, Second Edition, Pearson Education, 2007, ISBN 978-0131856035			

K.K.Wagh Institute of Engineering Education and Research, Nashik
(Autonomous from Academic Year 2022-23)

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: IV
COM222017: Advanced Data Structures Lab

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Practical : 04 hrs/week	**02**	**Term Work: 25 Marks** **Practical Exam: 50 Marks**

Prerequisite Courses:- COM222001:Fundamentals of Data structures, COM222003:Discrete Mathematics

Companion Course: COM222012: Advanced Data Structures

Course Objectives:

- To understand basic concepts of non linear data structures such as trees, graphs
- To study the concepts of hash table and files
- To learn advanced data structures such as indexing techniques and multiway search trees

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	**Bloom's Level**
CO1	Make use of non-linear data structures such as graph and trees to solve a given problem	3-Apply
CO2	Use different representations of symbol table	3-Apply
CO3	Apply the hash table and it's collision resolution methods and different file handling techniques	3-Apply
CO4	Use efficient indexing techniques and multiway search trees to store and maintain data	3-Apply
CO5	Analyze an algorithm used for solving a given problem	4-Analyze

List of Laboratory Experiments / Assignments

Sr. No.	**Laboratory Experiments / Assignments**	**CO Mapped**
1	**Flight management**: There are flight paths between cities. If there is a flight between city A and city B then there is an edge between the cities. The cost of the edge can be the time that flight takes to reach city B from A, or the amount of fuel used for the journey. Write a menu driven C++ program to represent this as a graph using adjacency matrix and adjacency list. The node can be represented by the airport name or name of the city. Check whether cities are connected through flight or not. Compare the storage representation.	CO1, CO5
2	**Graph traversal**: The area around the college and the prominent landmarks of it are represented using graphs. Write a menu driven C++ program to represent this as a graph using adjacency matrix /list and perform DFS and BFS.	CO1, CO5

3	**Activity on vertex(AOV) network**: Sandy is a well organized person. Every day he makes a list of things which need to be done and enumerates them from 1 to n. However, some things need to be done before others. Write a C++ code to find out whether Sandy can solve all his duties and if so, print the correct order	CO1, CO5
4	**Binary search tree:** Write a menu driven C++ program to construct a binary search tree by inserting the values in the order give, considering at the beginning with an empty binary search tree, After constructing a binary tree- i. Insert new node, ii. Find number of nodes in longest path from root, iii. Minimum data value found in the tree iv. Search a value v. Print values in ascending and descending order	CO1, CO5
5	**Expression tree**: Write a menu driven C++ program to construct an expression tree from the given prefix expression eg. +--a*bc/def and perform following operations: 1. Traverse it using post order traversal (non recursive) 2. Delete the entire tree 3. Change a tree so that the roles of the left and right pointers are swapped at every node	CO1, CO5
6	**A Dictionary using BST**: A Dictionary stores key and value pairs Data: Set of (key, value) pairs, Keys are mapped to values, Keys must be comparable, Keys must be unique. Standard Operations: Insert(key, value), Find(key), Delete(key) Write a menu driven C++ program to provide above standard operations on dictionaries and provide a facility to display whole data sorted in ascending/ Descending order. Also find how many maximum comparisons may require for finding any keyword. Use Binary Search Tree for implementation	CO1, CO5
7	**Tree using traversal sequence**: Write a C++ program to construct the binary tree with a given preorder and inorder sequence and Test your tree with all traversals	CO1, CO5
8	**A Dictionary using AVL**: A Dictionary stores key and value pairs Data: Set of (key, value) pairs, Keys are mapped to values, Keys must be comparable, Keys must be unique. Standard Operations: Insert(key, value), Find(key), Delete(key) Write a menu driven C++ program to provide above standard operations on dictionaries and provide a facility to display whole data sorted in ascending/ Descending order. Also find how many maximum comparisons may require for finding any keyword. Use Height balanced tree(AVL) and find the complexity for finding a keyword	CO2, CO5
9	**Telephone book management**: Consider the telephone book database of N clients. Write a menu driven C++ program to make use of a hash table implementation to quickly look up a client's telephone number. Use of two collision handling techniques and compare them using number of comparisons required to find a set of telephone numbers	CO3, CO5

10	**A Dictionary using Hash table**: A Dictionary stores key and value pairs Data: Set of (key, value) pairs, Keys are mapped to values, Keys must be comparable, Keys must be unique. Standard Operations: Insert(key, value), Find(key), Delete(key) Write a menu driven C++ program to provide above standard operations on dictionaries Write a menu driven C++ program to provide all the functions of a dictionary (ADT) using hashing and handle collisions using chaining.	CO3, CO5
11	**Sequential File:** The students' club members (MemberID, name, phone, email) list is to be maintained. The common operations performed include these: add member, search member, delete member, and update the information. Write a menu driven C++ program that uses file operation to implement the same and perform all operations.	CO3, CO5
12	**Min/max Heaps**: Marks obtained by students of second year in an online examination of a particular subject are stored by the teacher. Teacher wants to find the minimum and maximum marks of the subject. Write a menu driven C++ program to find out maximum and minimum marks obtained in that subject using heap data structure. Analyze the algorithm	CO4, CO5
13	**A Dictionary using STL map and Hashmap**: Implement Dictionary (key and value pairs) using using **STL map in C++** and **Hashmap in Java** and compare all dictionary implementation 1. BST 2. AVL 3. User defined Hash table 4. STL Map 5. Hashmap in Java Use Visual C++ and Java Compiler	CO1, CO2, CO3, CO5
14	Study Assignment: 1. Explain Data structures used in whatsapp in details 2. Consider following real time application and explain in detail the combinations of data structures and algorithms used in it. Social media applications require efficient and scalable data structures to manage user-generated content, facilitate user interactions, and ensure the reliability and availability of the platform. The primary challenge in designing data structures for social media applications is to accommodate the massive volume of data generated by users, while providing fast and responsive access to that data. Some specific challenges that data structures in social media applications must address include: • Handling user interactions such as likes, comments, and shares, and ensuring the integrity and consistency of those interactions. • Supporting fast and flexible search and filtering of content based on user preferences, geographic location, hashtags, and other criteria. • Managing relationships between users, such as friends, followers, and groups, and providing fast and efficient access to that information.	CO1 to CO5

Mini Project

	Student has to perform one mini project based on concepts covered in the course, Write a detailed problem statement for your project, Design and implement a code for the same using appropriate data Structures.	CO1 to CO5

Additional Programming Problems

1	**Skip Lists:** Write a C++ program to create a skip list for a given set of elements. Find the element in the set that is closest to some given value. (note: Decide the level of element in the list Randomly with some upper limit)	CO3, CO5
2	**Huffman algorithm**: Write a C++ program to implement a file compression algorithm that uses a binary tree. Your program should allow the user to compress and decompress messages containing alphabets using the standard Huffman algorithm for encoding and decoding.	CO1, CO5
3	**Tour management**: Tour operators organize guided bus trips across Maharashtra. Tourists may have different preferences. Tour operators offer a choice from many different routes. Every day the bus moves from starting city S to another city F as chosen by the client. On this way, the tourists can see the sights alongside the route traveled from S to F. Clients may have preference to choose the route. There is a restriction on the routes that the tourists may choose from, the bus has to take a short route from S to F or a route having one distance unit longer than the minimum distance. Two routes from S to F are considered different if there is at least one road from a city A to a city B which is part of one route, but not of the other route. Write a C++ program to solve above problem.	CO1, CO5
4	**Optimal Binary search tree**: Given sequence k = k1 <k2 < ... <kn of n sorted keys, with a search probability pi for each key ki . Write a C++ program to build the Binary search tree that has the least search cost given the access probability for each key.	CO2, CO5
5	**Trie** : Write a C++ program to store a collection of strings that have to be inserted in the trie and perform search operation	CO4, CO5

Guidelines for Laboratory Conduction

Use of coding standards and Hungarian notation, proper indentation and comments.

Use of open source software is to be encouraged.

Operating System recommended:- Linux or its derivative

Programming tools recommended: - Open Source line gcc/g++ (Visual C++ compiler for few assignments and note the difference)

Guidelines for Student's Lab Journal

The laboratory assignments are to be submitted by students in the form of a journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, problem statement, theory concepts in brief, algorithm, flowchart, test cases and conclusions). Program codes with sample outputs shall be submitted in soft form.

Guidelines for Termwork Assessment

Continuous assessment of laboratory work shall be based on the overall performance of a student.
Assessment of each laboratory assignment shall be based on rubrics that include
R1- timely completion (10),
R2- understanding of assignment (10) and
R3- presentation/clarity of journal writing (10) (Coding standard, Indentation, Hungarian notation, input validation etc)
Mini Project assessment will be based on Teamwork, Communication skill, Social relevance of mini project, Ethics followed.

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: IV
COM222018: Operating Systems Laboratory

Teaching Scheme:	Credit Scheme:	Examination Scheme:
Practical : 02 hrs/week	**01**	**Term Work: 25 Marks** **Practical Exam : 25 Marks**

Prerequisite Courses:- COM222001:Fundamentals of Data Structures

Companion Course:- COM222013: Operating Systems

Course Objectives:

- To understand operating system services, types of operating systems and shell scripts
- To study process scheduling algorithms and multithreading techniques
- To get acquainted with the concepts of synchronization, deadlock prevention and avoidance algorithms
- To learn concepts of memory management and I/O management techniques
- To introduce Linux operating systems

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	Bloom's Level
CO1	Explain operating system services, types of operating systems and basic shell commands	2- Understand
CO2	Illustrate the concept of process scheduling algorithms to solve scheduling problems	2- Understand
CO3	Compare algorithms for deadlock detection, prevention and avoidance	2- Understand
CO4	Use algorithms for page replacement and I/O management	3- Apply
CO5	Describe Linux commands and utilities such as grep, tr, sed, awk	2- Understand

Sr. No.	List of Laboratory Assignments/ Experiments	COs Mapped
1	Write a shell script for implementation of control flow statements.	**CO1**
2	Write a shell script to find factorial of a given number.	**CO1**
3	Write a C program to compute and print the average waiting time, average turnaround time and CPU burst times for the given list of processes. Display/print the Gantt chart for first come first serve, shortest job first, priority scheduling and round robin scheduling algorithm.	**CO2**
4	Write a C program to implement inter process communication using shared memory, pipes, named pipes and signals	**CO2**
5	Write a C program to implement producer-consumer problem	**CO3**
6	Write a C program to implement page replacement algorithms such as first in first out, least recently used and optimal page replacement	**CO4**
7	Installation of Linux operating system and basic configuration.	**CO5**
8	Assignment on Unix basic commands such as pwd, ls, cat, redirection and pipes and Unix utilities like tr, sed, grep, egrep, awk.	**CO5**

9	Execute following AWK operations on the text file : 1 Print the lines which match the given pattern. 2 Splitting a Line Into Fields 3 To find the length of the longest line present in the file 4 Printing the lines with more than specified characters	**CO5**
Guidelines for Laboratory Conduction		
Use of coding standards and Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. Operating System recommended: Linux or its derivative. Programming tools recommended: Open Source line gcc/g++		
Guidelines for Student's Lab Journal		
The laboratory assignments are to be submitted by students in the form of a journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, problem statement, theory concepts in brief, algorithm, flowchart, test cases and conclusions). Program codes with sample outputs shall be submitted in soft form.		
Guidelines for Term work Assessment		
Continuous assessment of laboratory work shall be based on overall performance of a student. Assessment of each laboratory assignment shall be based on rubrics that include R1- timely completion (10), R2- understanding of assignment (10) and R3- presentation/clarity of journal writing (10).		

S. Y. B. Tech. Computer Engineering
Pattern 2022 Semester: IV
COM222019: Microprocessor Lab

Teaching Scheme:	**Credit Scheme:**	**Examination Scheme:**
Practical :02 hrs/week	**01**	**Term Work: 25 Marks** **Practical Exam: 25 Marks**

Prerequisite Course:- COM22204: Digital Electronics and Logic Design

Companion Course:- COM222014: Computer Architecture

Course Objectives:

- To learn assembly language programming for Pentium microprocessor family
- To study system calls

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	**Bloom's Level**
CO1	Develop assembly language programs for Pentium microprocessor family	3-Apply
CO2	Make use of "syscall" to perform Input/output operations	3-Apply
CO3	Compare Near and Far procedures	2-Understand
CO4	Make use of instruction set, data types, assembler directives, flags to write assembly language programs	3-Apply
CO5	Illustrate looping and control structures using assembly language programs	2-Understand
CO6	Develop assembly language programs for strings	3-Apply

List of Laboratory Experiments / Assignments

Sr. No.	**Laboratory Experiments / Assignments**	**COs Mapped**
1.	Write an assembly language program to display "Hello World"	CO1, CO2, CO4
2.	Write an assembly language program to count positive and negative numbers from an array	CO1, CO2, CO4, CO5
3.	Write an assembly language program to find factorial of a given number	CO1, CO2, CO4, CO5
4.	Write an assembly language program to search an item in given data	CO1, CO2, CO4, CO5
5.	Write an assembly language program to sort 10 unsigned integers using Bubble Sort	CO1, CO2, CO4, CO5
6.	Write an assembly language program to find average of numbers declared in data segment	CO1, CO2, CO4, CO5
7.	Write an assembly language program to perform multiplication of two numbers	CO1, CO2, CO4, CO5
8.	Write an assembly language program to perform overlap and non overlap block transfer	CO1, CO2, CO4, CO5
9.	Write an assembly language program to accept a string and display its length	CO1, CO2, CO4, CO5, CO6
10.	Write an assembly language program to implement far procedure to concatenate two string	CO1, CO2, CO3, CO4, CO5, CO6

Guidelines for Laboratory Conduction

1. Teacher will brief the given experiment to students its procedure and outcome of this experiment. 2. Students will perform the allotted experiment individually under the supervision of faculty and lab assistant. 3. After performing the experiment students will check their output from the teacher. 4. After checking they have to write the conclusion of the final result.
Guidelines for Student's Lab Journal
Write-up should include title, aim, procedure, results and conclusion.
Guidelines for Termwork Assessment
Continuous assessment of laboratory work shall be based on the overall performance of a student. Assessment of each laboratory assignment shall be based on rubrics that include R1- timely completion (10), R2- understanding of assignment (10) and R3- presentation/clarity of journal writing (10)

S. Y. B. Tech. Computer Engineering **Pattern 2022 Semester: IV** **COM222020: Project Based Learning –Client Side Technology**		
Teaching Scheme:	**Credit Scheme:**	**Examination Scheme:**
Practical : 02 hrs/week	**01**	**Term Work : 25 Marks**

Prerequisite Courses :- COM222006: Design Thinking

Companion Course:- COM222016: Client Side Technology

Course Objectives:

- To develop critical thinking and problem solving ability by exploring and proposing solutions to realistic/social problem
- To apply alternative approaches for selecting client side technologies
- To emphasizes learning activities that are long-term, inter-disciplinary and student-centric
- To provide every student the opportunity to get involved either individually or as a group so as to develop team skills and learn professionalism
- To develop an ecosystem that promotes entrepreneurship and research culture among the students through web based development environment

Course Outcomes: On completion of the course, students will be able to–

	Course Outcomes	**Bloom's Level**
CO1	Identify the real life and societal problem	3-Apply
CO2	Build web pages using client side technologies	3-Apply
CO3	Make use of Angular for web development	3-Apply
CO4	Make use of front-end frameworks for web development	3-Apply

List of Tasks

Sr. No.	**Tasks**	**CO Mapped**
1	Creating teams, assigning roles and responsibilities for project based learning	**CO1**
2	Brain storming: Ideation, setting actionable problem statement, identify stakeholders, people/ organization, problems and opportunities, prepare questionnaire and discuss with stakeholders	**CO1**
3	**Use suitable Client Side Technology to design and develop mini project**	
3.1	Design and develop GUI using client side technologies Hint: HTML,CSS	**CO2**
3.2	Update task 3.1 using Java Script to apply dynamic behavior Hint: Java Script	**CO2**

3.3	Rebuild task 3.2 into a single page application using Angular	**CO3**
3.4	Redesign task 3.3 and develop dynamic application using Node.JS and React	**CO4**

Guidelines for Laboratory Conduction

Client Side Technology (MOOC) is companion course for Project-Based Learning (PBL) - Client Side Technology. PBL is an instructional approach designed to give students an opportunity to develop knowledge and skills through engaging projects set around challenges and problems they may face in the real world. It is more than just projects. With these, students investigate and respond to an authentic, engaging, and complex problem and providing feasible solution using client side web technologies. It requires mentoring by faculty throughout the semester for successful completion of the project tasks selected by the students per batch. The batch should be divided into sub-groups of 4 to 5 students. Idea presentation and implementation under this course is carried throughout the semester and evaluation is done on the basis of internal continuous assessment.

Guidelines for Student's Lab Journal

The laboratory tasks are to be completed by students in the form of a report. Report consists of Certificate, table of contents, title, team structure, surveys conducted, problem statement, use cases, concepts in brief, conclusions. A mini project shall be presented in the soft form and report shall be submitted to the mentor for evaluation.

Guidelines for Term work Assessment

It is recommended that all activities should be recorded regularly, regular assessment of work need to be done and proper documents need to be maintained by both students as well as mentor. Continuous Assessment Record is to be maintained by all mentors.

Recommended rubrics for weekly assessment / evaluation:

Task 1 : Creating teams, assigning roles and responsibilities for project based learning Task 2 : Ideation	10 M
Task 3.1: Design and develop GUI using client side technologies	15 M
Task 3.2: Update task 3.1 using Java Script to apply dynamic behavior	15 M
Task 3.3: Rebuild task 3.2 into a single page application using Angular	15 M
Task 3.4: Redesign task 3.3 and develop dynamic application using Node.JS and React	15 M
Report Writing	30 M
Task 3.1, 3.2, 3.3, 3.4 : 15 marks each (R1 : Timely completion and R2: Implementation)	