Semester: 4	
Programme: Data Science	
Course: Discrete Structures	
Paper code:	**Credits: 4**
Hours/week: 4	
Category: Core/MDC/SEC/VAC: Core	
Theory / Practical / Composite: Theory	
No of Module: 1	

Course Outcome:

1. **Remember** fundamental definitions, terminology, and principles related to counting theory, propositional logic, and graph structures.
2. **Understand** the conceptual framework of recurrence relations, generating functions, and the properties of logical connectives and well-formed formulas.
3. **Apply** combinatorial analysis, mathematical induction, and specific graph algorithms, such as Dijkstra's and Kruskal's, to solve discrete mathematical problems.
4. **Analyze** the behavior of recurrence relations using the Master Theorem and the structural properties of various graphs, including connectivity and planarity.
5. **Evaluate** the validity of mathematical arguments using tautologies and normal forms, and judge the efficiency of different graph algorithm implementations.
6. **Create** mathematical models and algorithmic solutions by synthesizing concepts from graph theory and recurrence relations to address complex data science problems

SYLLABUS

UNIT/Module	CONTENT	HOURS or NUMBER OF CLASSES	CO Mapping	COGNITIVE LEVEL
1.	**Counting Theory**: Pigeonhole Principle; Mathematical Induction; Principle of Inclusion and Exclusion.	**8**	**CO1, CO3**	**K1, K3**
2.	**Recurrence Relations**: Formulation, Substitution Method, Linear Recurrence Relations with constant coefficients and their solution, Generating functions, Recurrence Trees, Master Theorem.	**9**	**CO2, CO4, CO6**	**K2, K4, K6**
3.	**Prepositional Logic**: Logical Connectives, Well-formed Formulas, Tautologies, Equivalences, Normal Forms.	**7**	**CO1, CO5**	**K1, K5**
4.	**Graph Theory:** Definition of Graph, Graph Terminology, Finite and Infinite graphs. Directed and undirected graphs, Degree, Isolated vertex, Pendant vertex. Null graphs. Walks: Paths and circuits. Connected and disconnected graphs, Euler's graphs, Hamiltonian paths and circuits. Planar	**13**	**CO1, CO4, CO6**	**K1, K4, K6**

	Graph, Isomorphic graph, Graph Coloring Problem			
5.	**Graph Algorithms**: Graph representation, Floyd's shortest Path algorithm, Trees, Minimal Spanning Tree using Kruskal Algorithm and Prim's Algorithm, Dijkstra's Algorithm, Breadth First Search (BFS), Depth First Search (DFS), Connected components.	**15**	**CO3, CO5, CO6**	**K3, K5, K6**

Text Books

1. C.L. Liu, D.P., Elements of Discrete mathematics, 2nd Edition, Tata McGraw Hill, 1985
2. Kenneth Rosen, Discrete Mathematics and Its Applications, Sixth Edition, McGraw Hill 2006
3. M. O. Albertson and J. P. Hutchinson, Discrete Mathematics with Algorithms, John wiley Publication, 1988
4. J. L. Hein, Discrete Structures, Logic, and Computability, 3rd Edition, Jones and Bartlett Publishers, 2009
5. D.J. Hunter, Essentials of Discrete Mathematics, Jones and Bartlett Publishers

Evaluation	Theory CIA: 30 Semester Exam: 70 Total: 100	
Paper Structure for Theory Semester Exam Module:	Short questions (5 marks each)	Long questions (15 marks each)
	5 out of 7	3 out of 5

Course outcomes (COs) and Cognitive Level Mapping

COs	CO Description	Cognitive levels
CO1	**Remember** fundamental definitions, terminology, and principles related to counting theory, propositional logic, and graph structures.	K1
CO2	**Understand** the conceptual framework of recurrence relations, generating functions, and the properties of logical connectives and well-formed formulas.	K2
CO3	**Apply** combinatorial analysis, mathematical induction, and specific graph algorithms, such as Dijkstra's and Kruskal's, to solve discrete mathematical problems.	K3
CO4	**Analyze** the behavior of recurrence relations using the Master Theorem and the structural properties of various graphs, including connectivity and planarity.	K4
CO5	**Evaluate** the validity of mathematical arguments using tautologies and normal forms, and judge the efficiency of different graph algorithm implementations.	K5
CO6	**Create** mathematical models and algorithmic solutions by synthesizing concepts from graph theory and recurrence relations to address complex data science problems	K6