MTH 202 Discrete Mathematics and Its Applications

Module Objectives:

To understand the concepts: Mathematical Reasoning, Combinatorial Analysis, Discrete Structures, Algorithmic Thinking, and Applications.

Contents:

Logic and Proof, Algorithms, the Integers, Mathematical Reasoning, Induction, and Recursion, Counting, Relations and functions, Graphs, Trees.

Detailed Course

1 The Foundations: Logic and Proof,

1.1 Logic

- Propositions, Proposition variables, Truth table, conjunction, disjunction, Exclusive, implications, converse, inverse, Contra positive, Bi-conditional, Tautology, Contradiction, translating English sentences, logic and bit operations

1.2 Propositional Equivalences

- Introduction, Logical equivalences: Identity law, Domination law, Idempotent laws, Double negation law, commutative law, associative law, Distributive law, De-Morgan's law, Absorption law, Negation law (Verification)

1.3 Brief introduction and examples of Predicates and Quantifiers

1.4 Methods of Proof

Methods of proving theorems (direct proofs, indirect proofs, vacuous and trivial proofs, proof by contradiction).

2 The Fundamentals: Algorithms, the Integers, and Matrices

2.1 Algorithms

Introduction, searching algorithms (linear, binary), sorting (bubble, insertion), greedy algorithms, halting problem

2.2 The Growth of Functions

Introduction, big-O notation, the growth of combinations of functions, big-omega and big-theta notation

2.3 Complexity of Algorithms

Introduction, time complexity, worst case complexity, average case complexity, understanding the complexity of algorithms

2.4 The Integers and Division

Introduction, division, primes, the fundamental theorem of arithmetic, the infinitude of primes, the division algorithm, GCD and LCM, modular arithmetic, applications of congruence's, Cryptology.

3. Mathematical Reasoning, Induction, and Recursion

3.1 Sequences and Summations

Introduction, sequences, recurrence relations, special integer sequences, summations

3.2 Mathematical Induction

Introduction, mathematical induction, Recursive Definitions.

Introduction, recursively defined function,

3.3 Recursive Algorithms, recursion and iteration, the merge sort

4. Counting

4.1 Basic counting principle – The sum rule and the product rule.

4.2 Permutation of n different objects, The number of r – permutations of n distinct objects when (a) repetition of objects are not allowed (b) repetition of objects are allowed.

Permutations of n objects when the things are not distinct, circular permutations.

Restricted permutations – The number of r-permutations of n different objects in which (i) k particular objects do not occur and (ii) k particular objects are always present.

4.3 Combination: - r-combinations of n different objects Restricted combinations, combinations with repetitions: the number of combinations of n objects taken r at a time with repetition is c(n+r-1, r)

4.4Binomial Theorem, Binomial coefficients and Pascal triangle Pascal's identity.

4.5 The pigeonhole principle and Inclusion and Exclusion principle.

4.6Recurrence relation and solving it.

5. Relations and Functions

5.1 Product sets, Binary relations, Domain and Range of binary relation.

5.2 Types of relations – Inverse relation, Identity relation, universe relations, void relation, complementary relation, ternary relation and n-ary relation.

5.3 Representation of relations – Table of relation, Arrow diagrams of relation, Graph of relation, Matrix of relation, Directed graph of a relation on a set A.

5.4 Boolean matrix, Boolean matrix operation, Boolean product of two matrices, complement of Boolean matrix.

5.5 Properties of relations – reflexive, irreflexive, symmetric, asymmetric, anti-symmetric and transitive relations. Equivalence relation, Equivalence relation and partition, Equivalence classes and quotient set. Partial order relation, Partial ordered set

5.6 Composition of two relations, matrix of composition relations properties –

a) If R is a relation from A to B and S a relation from B to C, then $M_{SOR} = M_R$ ¤ $M_{S.}$*(without proof)*

b) If R is a relation from A to B, S a relation from B to C, and T a relation from C to D, then To (S o R) = (T o S) o R. *(without proof)*

c) Let A, B and C be sets, R a relation from A to B, and S a relation from B to C. Then $(S \circ R)^{-1} = R^{-1} \circ S^{-1}$ *(without proof)*

5.7 Concept of function, Domain and Range, image and pre-image, Graph of a function $f : A \to B$, Equality of functions, Real valued function, constant function and Identity function. Special functions – Floor function, ceiling function

5.8 Types of functions – onto function, one-to-one function, one-to-one correspondence between A and B, Inverse function.

5.9 The composition of two functions, Properties – (a) $I_B o f = f$, (b) $f o I_A = f$, (c) $f^{-1} o f = I_A$, (d) $f o f^{-1} = I_B$, *(with proof)* , (f) $(g o f)^{-1} = f^{-1} o g^{-1}$.

6. Graphs

6.1 Introduction to Graphs and graph terminologies:

Simple graph, multiple graph and pseudo graph, order of a graph and size of a graph adjacent vertices, adjacent edges, degree of a vertex, isolated vertex and Pendant vertex. Degree sequence of a graph.

Properties (with proofs):

a) The sum of the degree of the vertices of a graph is equal to twice the number of edges.
b) The number of odd vertices in a graph is always even.

Special types of simple graph – Isolated graph, complete graph, Regular graph, Path graph, Cycle graph, Wheel graph, Bipartite graph and complete bipartite graph, Graphs of regular Platonic Solids.

Properties (with proofs):-

a) The total number of edges in a complete graph Kn is $\frac{n(n-1)}{2}$
b) The number of vertices in a r-regular graph is even if r is odd.
c) The complete graph K_n is the regular graph of degree $n – 1$.
d) In the cyclic graph C_n, size of C_n is equal to order of C_n.
e) The size of wheel W_n is twice the size of C_n.
f) The sum of the degrees of vertices in W_n is four times the size of C_n.
g) Size of the complete bipartite graph Km, n is $m \times n$ and order is $m + n$.

6.2 Representing Graphs: Adjacency list, Adjacency matrix, and Incidence matrix.

6.3 Isomorphism of Graph: Isomorphic graphs, Isomorphism classes, Self Complementary.

6.4 Connectivity: walk, trial and circuit, Path and Cycle, Connected graph, Cut-sets and Cut-vertices. Edge connectivity and vertex connectivity.

6.5 Euler and Hamilton Paths:

Eulerian trial, Eulerian Circuit, Eulerian graph, Konigsberg Bridge problem. Theorems *without proofs)*:- a. A co

b. A connected graph G has Eulerian trial if and only if it has exactly two odd vertices.

path, Hamiltonian cycle and Hamiltonian graph.

Theorems *(without proofs)*

a) (Ore's) A connected graph with n vertices is Hamiltonian if for any two non-adjacent vertices u and v, deg (u) + deg (v)≥n.

b) (Dirac) A connected graph with n(>2) vertices is Hamiltonian if degree of every vertex is at least $^n/_2$.

Labeled graphs and weighted graphs,

6.6 Shortest-Path Problems: – Dijkstra's algorithm

6.7 Digraph, Simple digraph, Reflexive, Symmetric and Transitive digraph, Loop and parallel arc (edge), adjacent vertices and degree of vertices, Source vertex and Sink vertex.

Theorem (without proof) – In a digraph, the sum of the in-degrees of vertices, the sum of the out-degrees of vertices and the number of edges are equal to each other.

6.8 Representation of digraph - Adjacency list, Adjacency matrix and Incidence matrix.

6.9 Connectivity of digraphs – underlying graph, directed walk, closed walk, directed path, directed cycle, spanning path. Weakly connected, unilaterally connected and strongly connected theorems (without proofs):

a) A diagraph D is unilaterally connected if it has a spanning path in D.

b) A diagraph D is strongly connected if it has a closed spanning path in D.

7. Trees

7.1 Introduction, rooted tree, non-rooted tree, root vertex, Terminal vertex, Internal vertex, Level of a vertex, H

7. 2 Properties of tree (with proofs).

a) Let G(V, E) be a loop-free undirected graph. Then G is a tree if there is a unique path between any two vertices of G.

b) A tree with n vertices has exactly n – 1 edges.

c) In any tree G, there are at least two pendant vertices.

d) A forest G with n vertices has n – k edges, where k is the number of components of G.

7.3 Spanning tree and Methods of constructing a spanning tree from a graph by

a) Breadth – first search and b) Depth – first search (Backtracking), Determination of the number of spanni

7.4 Minimum spanning tree – a) Kruskal algorthm b) Prim's algorithm.

7.5 Tree Traversal: In order, Pre-order, and post order traversal

7.6 Applications of Trees: Binary expression tree

7.7 Full binary tree and its properties:

a) The number of vertices n in a binary tree is always odd.

b) The number of pendant vertices of a binary tree with n vertices is ½ (n + 1).

c) The number of internal vertices in a binary tree is one less than the number of pendant vertices.

d) The maximum number of vertices possible in K-level binary tree is $2^0 + 2^1 + 2^2 + .. + 2^K \geq n$.

e) The minimum possible height of an n-vertex binary tree is min l_{max} = $\lceil \log_2 (n+1) - 1 \rceil$, where L_{max} = max^m level of any vertex.

f) The maximum possible height of an n – vertex binary tree is max $l_{max} = \frac{(n-1)}{2}$

Lecture: 48 Hours

Tutorial: 12 Hours

Text Book

Rosen K.H**.,** ***Discrete Mathematics and its applications, 5th Edition,*** *McGraw Hill Companies*

References

Kolma, Busby, Ross; ***Discrete Mathematical Structures,*** Prentice – Hall of India.

R. Joshnsonbaugh; ***Discrete Mathematics,*** Pearson Education Asia.

Seymour Lipschutz and Marc Lipson; ***Discrete Mathematics,*** (Schaum's Outline).

S.M. Maskey: ***First course in Graph Theory,*** Published by Ratna Pustak Bhandar.

E. G. Gooduire and M. M. Paramenter, ***Discrete mathematics with graph theory***, Prentice – Hall of India.

Narsingh Deo: ***Graph Theory (with application to engineering and computer science)***, Prentice – Hall of India Pvt. Ltd.