R18

Code No: 153CF

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

B. Tech II Year I Semester Examinations, April/May -2023

DISCRETE MATHEMATICS

(Common to CSE(DS),CSE(IOT), AI&DS, AI&ML, CSD)

Time: 3 HoursMax. Marks: 75

Note: i) Question paper consists of Part A, Part B.
ii) Part A is compulsory, which carries 25 marks. In Part A, answer all questions.
iii) In Part B, Answer any one question from each unit. Each question carries 10 marks and may have a, b as sub questions.

PART – A

(25 Marks)

A. Define Quantifiers. [2]

B. Show that (p 🎭 r) ☂ (q 🎭 r) and (p ☂ q)🎭 r logically equivalent. [3]

C. Write properties of lattice. [2]

D. Write applications of n-array relations. [3]

E. Define structural induction. [2]

F. Compare well and partial ordering principle. [3]

G. How to calculate the coefficient of generating function? [2]

H. Find the recurrence relation of the sequence S(n) = a^n, n 💿 1. [3]

I. Mention applications of a tree. [2]

J. Define a complete bipartite graph with an example. [3]

PART – B

(50 Marks)

2.a) Discuss an important list of implications and equivalences withexamples.

b) Show that R is logically derived from P → Q, Q → R, and P [5+5]

OR

3.a) Show that 👓x P(x) ☁ 👓x Q(x) and 👓x (P(x) ☁ Q(x)) are not logically equivalent.

b) Show that the following premises are inconsistent.

i If Jack misses many classes through illness, then he fails high school.
ii. If Jack fails high school, then he is uneducated.
iii. If Jack reads a lot of books, then he is not uneducated.
iv. Jack misses many classes through illness and reads a lot of books.[5+5]

4.a) Let R be a group of all real numbers under addition and R+ be a group of all positive real numbers under multiplication. Show that the mapping f :

R+ → R defined by f(x) = log10 x for all x ϵ R is an isomorphism.

b) Show that the function f: Q → Q defined by f(x) = 2x + 3 is both one-one and onto. Here Q is the set of all rational numbers. [5+5]

OR

5.a) Illustrate the properties of relations with suitable examples.

b) If M is a set of all non singular matrices of order n x n, then show that M is a group with a request to matrix multiplication. Is (M, *) an abelian group? Justify your answer. [5+5]

6.a) Define Algorithms. Write an algorithm to check whether an element belongs to an array or not. Find complexity of algorithm.

b) Discuss in detail Pigeonhole principle with an example. [5+5]

OR

7.a) Find the number of integers between 1 and 250, that are divisible by any of the integers 2, 3, 5, and 7.

b) State the strong induction. Prove that a positive integer > 1 is either a prime number or it can be written as a product of prime numbers. [5+5]

8.a) Solve the non-homogenous recurrence relation Sn = 3Sn-1 + 10 Sn-2 +7.5, S0 =4 and S1 = 3.

b) Use the method of generating function to solve the recurrence relation Sn + 3 Sn-1 – 4Sn-2 = 0 where n ≥ 2 S0=3, S1 = -2. [5+5]

OR

9.a) How many onto functions are there from a set with six elements to a set with three elements?

b) Discuss in detail about Linear Non-homogeneous Recurrence Relations with Constant Coefficients with an example. [5+5]

10.a) Discuss in detail about tree traversal techniques with an example.

b) Define Spanning tree and Minimum cost Spanning tree. State and discuss Kruskal's algorithm with an example. [5+5]

OR

11.a) Give an example of a graph which is

i. Eulerian but not Hamilton,
ii. Hamilton but not Eulerian,
iii. Eulerian and Hamilton, and iv. Neither Hamilton nor Eulerian.

b) Draw the complete graph K5 with vertices A, B, C, D, and E and draw all complete sub- graph of K5 with 4 vertices. [5+5]

---ooOoo—

Answer Key

PART - A

1. Define Quantifiers:
Quantifiers are logical symbols that specify the quantity of specimens in the domain of discourse for which a predicate is true. The two main types are the universal quantifier ($\forall$), meaning "for all," and the existential quantifier ($\exists$), meaning "there exists."

2. Show that $(p \rightarrow r) \vee (q \rightarrow r)$ and $(p \vee q) \rightarrow r$ are logically equivalent:
The statements $(p \rightarrow r) \vee (q \rightarrow r)$ and $(p \vee q) \rightarrow r$ are logically equivalent. Proof: $(p \rightarrow r) \vee (q \rightarrow r)$ simplifies to $(\neg p \vee r) \vee (\neg q \vee r)$, which simplifies to $\neg (p \vee q) \vee r$, equivalent to $(p \vee q) \rightarrow r$.

3. Write properties of lattice:
A lattice is an algebraic structure with two binary operations (meet $\wedge$ and join $\vee$) satisfying commutativity, associativity, absorption, and idempotence.

4. Write applications of n-array relations:
N-ary relations are used in databases to represent relationships between entities, in formal semantics to describe relationships between linguistic elements, and in graph theory to describe hypergraphs.

5. Define structural induction:
Structural induction is a proof technique used to prove properties of recursively defined structures, such as trees or lists, by proving the base case and the inductive step.

6. Compare well and partial ordering principle:
A well-ordering is a total order where every non-empty subset has a least element. A partial ordering is a binary relation that is reflexive, antisymmetric, and transitive, but not necessarily total.

7. How to calculate the coefficient of generating function:
To calculate the coefficient of a generating function $G(x)$, expand $G(x)$ as a power series and identify the coefficient of the desired term x^n.

8. Find the recurrence relation of the sequence $S(n) = a^n$, $n \geq 1$:
The recurrence relation for the sequence $S(n) = a^n$ is $S(n) = a \cdot$

$S(n-1)$ with $S(1) = a$.

9. Mention applications of a tree:
Trees are used in data structures (binary search trees, heaps), network routing algorithms, and hierarchical data representation (organizational charts, file systems).

10. Define a complete bipartite graph with an example:
A complete bipartite graph $K_{m,n}$ is a graph that can be divided into two disjoint sets of vertices such that every vertex in one set is connected to every vertex in the other set. Example: $K_{3,2}$ has sets $\{A,B,C\}$ and $\{1,2\}$ with edges $(A,1)$, $(A,2)$, $(B,1)$, $(B,2)$, $(C,1)$, $(C,2)$.

PART - B

2. Logical Implications and Equivalences
a) Discuss an important list of implications and equivalences with examples.
1. Implication (→):
$P \rightarrow Q$: If P is true, then Q must be true.
Example: If it rains (P), then the ground gets wet (Q).

2. Conjunction (∧):
$P \land Q$: Both P and Q are true.
Example: It is raining (P) and the ground is wet (Q).

3. Disjunction (∨):
$P \lor Q$: Either P or Q (or both) are true.
Example: It is raining (P) or it is snowing (Q).

4. Biconditional (↔):
$P \leftrightarrow Q$: P is true if and only if Q is true.
Example: The ground is wet (Q) if and only if it is raining (P).

5. Negation (¬):
$\neg P$: P is not true.
Example: It is not raining (¬P).

b) Show that R is logically derived from $P \rightarrow Q$, $Q \rightarrow R$, and P.

1. Given: $P \rightarrow Q$
2. Given: $Q \rightarrow R$
3. Given: P
4. From (1) and (3), Q is true.
5. From (2) and (4), R is true.
6. Thus, R is derived.

3. Logical Equivalence and Consistency

a) Show that $\exists x P(x) \wedge \exists x Q(x)$ and $\exists x (P(x) \wedge Q(x))$ are not logically equivalent.

1. $\exists x P(x) \wedge \exists x Q(x)$ means there exists some x for which P(x) is true and there exists (possibly different) y for which Q(y) is true.
2. $\exists x (P(x) \wedge Q(x))$ means there exists a single x for which both P(x) and Q(x) are true.
3. Example: P(x) = "x is a dog", Q(x) = "x is a cat". The first can be true if there exists at least one dog and at least one cat, while the second requires an x that is both a dog and a cat, which is not possible.

b) Show that the following premises are inconsistent.

1. If Jack misses many classes through illness, then he fails high school.
$M \rightarrow F$
2. If Jack fails high school, then he is uneducated.
$F \rightarrow U$
3. If Jack reads a lot of books, then he is not uneducated.
$R \rightarrow \neg U$
4. Jack misses many classes through illness and reads a lot of books.
$M \wedge R$

1. From (4), M is true and R is true.
2. From (1) and M, F is true.
3. From (2) and F, U is true.
4. From (3) and R, ¬U is true.
5. U and ¬U is a contradiction, thus the premises are inconsistent.

4. Group and Isomorphism

a) Show that the mapping $f : R^+ \rightarrow R$ defined by $f(x) = \log_{10} x$ for all $x \in R^+$ is an isomorphism.

1. **Homomorphism:** $f(xy) = \log_{10} (xy) = \log_{10} x + \log_{10} y = f(x) + f(y)$, preserving the group operation.
2. **Injective:** If $f(x) = f(y)$, then $\log_{10} x = \log_{10} y$, thus $x = y$.

3. Surjective: For any $r \in R$, there exists $x = 10^r \in R^+$ such that $f(x) = r$.
4. Therefore, f is an isomorphism.

b) Show that the function $f: Q \rightarrow Q$ defined by $f(x) = 2x + 3$ is both one-one and onto.
1. Injective: If $f(x) = f(y)$, then $2x + 3 = 2y + 3$, thus $x = y$.
2. Surjective: For any $q \in Q$, $f(x) = q$ implies $2x + 3 = q$. Solving for x, $x = \frac{q - 3}{2} \in Q$.
3. Thus, f is bijective (one-one and onto).

5. Properties of Relations and Groups
a) Illustrate the properties of relations with suitable examples.
1. Reflexive: R on set A if $(a, a) \in R$ for all $a \in A$. Example: $=$ on integers.
2. Symmetric: R on set A if $(a, b) \in R$ implies $(b, a) \in R$. Example: "is a sibling of".
3. Transitive: R on set A if $(a, b) \in R$ and $(b, c) \in R$ implies $(a, c) \in R$. Example: "is an ancestor of".

b) If M is a set of all non-singular matrices of order $n \times n$, then show that M is a group with respect to matrix multiplication. Is $(M, *)$ an abelian group? Justify your answer.
1. Closure: The product of two non-singular matrices is non-singular.
2. Associativity: Matrix multiplication is associative.
3. Identity: The identity matrix I is non-singular and $A \cdot I = A$.
4. Inverse: Every non-singular matrix A has an inverse A^{-1} such that $A \cdot A^{-1} = I$.
5. Not Abelian: In general, $AB \neq BA$ for non-singular matrices, so $(M, *)$ is not abelian.

6. Algorithms and Pigeonhole Principle
a) Define Algorithms. Write an algorithm to check whether an element belongs to an array or not. Find the complexity of the algorithm.
1. Definition: An algorithm is a step-by-step procedure for solving a problem or performing a task.
2. Algorithm:
```python
def element_in_array(array, element):
    for item in array:
        if item == element:
            return True
    return False
```

```

**3. Complexity:** The time complexity is $O(n)$, where $n$ is the number of elements in the array.

**b) Discuss in detail Pigeonhole principle with an example.**
**1. Pigeonhole Principle:** If $n$ items are put into $m$ containers, with $n > m$, then at least one container must contain more than one item.
**2. Example:** If you have 10 pairs of socks and 9 drawers, at least one drawer must contain more than one pair of socks.

**7. Number Theory and Induction**
**a) Find the number of integers between 1 and 250 that are divisible by any of the integers 2, 3, 5, and 7.**
**1. Use the Inclusion-Exclusion Principle:**
$N(2,3,5,7) = N(2) + N(3) + N(5) + N(7) - N(2,3) - N(2,5) - N(2,7) - N(3,5) - N(3,7) - N(5,7) + N(2,3,5) + N(2,3,7) + N(2,5,7) + N(3,5,7) - N(2,3,5,7)$.
**2. Calculate each term and sum accordingly:**

$$
N(2,3,5,7) = \left\lfloor \frac{250}{2} \right\rfloor + \left\lfloor \frac{250}{3} \right\rfloor + \left\lfloor \frac{250}{5} \right\rfloor + \left\lfloor \frac{250}{7} \right\rfloor - \left\lfloor \frac{250}{6} \right\rfloor - \left\lfloor \frac{250}{10} \right\rfloor - \left\lfloor \frac{250}{14} \right\rfloor - \left\lfloor \frac{250}{15} \right\rfloor - \left\lfloor \frac{250}{21} \right\rfloor - \left\lfloor \frac{250}{35} \right\rfloor + \left\lfloor \frac{250}{30} \right\rfloor + \left\lfloor \frac{250}{42} \right\rfloor + \left\lfloor \frac{250}{70} \right\rfloor + \left\lfloor \frac{250}{105} \right\rfloor - \left\lfloor \frac{250}{210} \right\rfloor = 125 + 83 + 50 + 35 - 41 - 25 - 17 - 16 - 11 - 7 + 8 + 5 + 3 + 2 - 1 = 183.
$$

**b) State the strong induction. Prove that a positive integer > 1 is either a prime number or it can be written as a product of prime numbers.**
**1. Strong Induction:** To prove $P(n)$ for all $n \geq k$, assume $P(k), P(k+1), \ldots, P(m)$ are true and use this to prove $P(m+1)$.
**2. Proof:**
**Base case:** $n = 2$, which is prime.
Inductive step: Assume every integer $k \leq n$ is prime or a product of primes. For $n+1$:
If $n+1$ is prime, done.
If not, $n+1 = ab$, with $1 < a, b < n+1$. By hypothesis, $a$ and $b$ are either prime or product of primes, so $n+1$ is a product of these primes.
```

8. Recurrence Relations

a) Solve the non-homogenous recurrence relation $S_n = 3S_{n-1} + 10 S_{n-2} + 7.5$, $S_0 = 4$ and $S_1 = 3$.

1. Solution:

Complementary solution: Solve $S_n = 3S_{n-1} + 10S_{n-2}$.

Particular solution: Guess $S_n = A$ for non-homogenous part.

Combine solutions and use initial conditions to find constants.

b) Use the method of generating function to solve the recurrence relation $S_n + 3S_{n-1} - 4S_{n-2} = 0$ where $n \geq 2$, $S_0 = 3$, $S_1 = -2$.

1. Solution:

Let $G(x) = \sum_{n=0}^{\infty} S_n x^n$.

Formulate generating function equation from recurrence relation.

Solve for $G(x)$ and extract coefficients.

9. Functions and Recurrence Relations

a) How many onto functions are there from a set with six elements to a set with three elements?

1. Use the formula for the number of onto functions:
$m^n - \sum_{k=1}^{m-1} \binom{m}{k} (m-k)^n$, where $m = 3$ and $n = 6$.

2. Calculate: $3^6 - \binom{3}{1}2^6 + \binom{3}{2}1^6 = 729 - 3 \cdot 64 + 3 \cdot 1 = 729 - 192 + 3 = 540$.

b) Discuss in detail about Linear Non-homogeneous Recurrence Relations with Constant Coefficients with an example.

1. Definition: A linear non-homogeneous recurrence relation has the form $a_n = c_1a_{n-1} + c_2a_{n-2} + \cdots + c_ka_{n-k} + f(n)$.

2. Example: $a_n = 2a_{n-1} + 3$ with initial conditions. Solve using the method of undetermined coefficients or variation of parameters.

10. Trees and Graphs

a) Discuss in detail about tree traversal techniques with an example.

1. Preorder Traversal: Visit root, traverse left subtree, traverse right subtree.
Example: For tree with root A, left child B, right child C: A, B, C.

2. Inorder Traversal: Traverse left subtree, visit root, traverse right subtree.
Example: B, A, C.

3. Postorder Traversal: Traverse left subtree, traverse right subtree, visit root.
Example: B, C, A.

b) Define Spanning tree and Minimum cost Spanning tree. State and discuss Kruskal's algorithm with an example.

1. Spanning Tree: A subgraph that includes all the vertices of the original graph and is a tree.

2. Minimum Cost Spanning Tree: A spanning tree with the smallest possible total edge weight.

3. Kruskal's Algorithm:

Sort all edges by weight.

Add edges to the spanning tree, ensuring no cycles, until all vertices are connected.

Example: Apply to a graph with vertices {A, B, C} and edges {A-B:1, B-C:2, A-C:3}.