Pioneering Excellence in Education, Research & Innovation

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY BANGALORE
Quiz 1 - Term I (2023-24)

Course Code: CS 201	Course Title: Discrete Mathematics
Name:	Roll Number:
Duration: 120 mins	Max Marks: 20

Good Luck!!

Instructions

a: You must write the answers *only in* the provided space. No additional answer sheet will be made available. You can continue answering a specific question in some other empty space with proper labelling and reference. No need to submit the rough work. You will not be provided additional question paper at any cost.

b: You should not call the instructor or the TA for any kind of clarification. It will invite negative markings.

c: You are allowed to carry only your handwritten notes. This is an open notes exam.

d: Any result proved in the class can be used (with proper citation) directly, without proving it.

e: Any assumption made while answering should be clearly stated. All questions carry equal marks. Breakdown for sub-parts is mentioned with the questions.

1. Solve the following (1 x 5).

   (a) Construct a truth table for the following:

$$\neg(p \to q) \lor (\neg p \land q)$$

   (b) Check for consistency : If the file system is not locked, then new messages will be queued. If the file system is not locked, then the system is functioning normally, and conversely. If new messages are not queued, then they will be sent to the message buffer. If the file system is not locked, then new messages will be sent to the message buffer. New messages will not be sent to the message buffer.

   (c) Check if the two expressions are equivalent :

   - $((p \lor q) \to (q \lor r))$
   - $(p \to q) \to r$

   (d) Check if this argument is valid or not : If you are at IIITB , then you are on Tinder. If you are not single , then you are not on Tinder. Therefore, if you are at IIITB, then you are single .

   (e) Check if this argument is valid or not : All IIITB students like football. Manda likes football. Therefore , Manda is a student of IIITB.

1) a)

p	q	$\sim p$	$\sim q$	$p \to q$	$\sim(p \to q)$	$\sim p \wedge q$	$\sim(p \to q) \vee (\sim p \wedge q)$
T	T	F	F	T	F	F	F
T	F	F	T	F	T	F	F
F	T	T	F	T	F	T	F
F	F	T	T	T	F	F	F

b) Locked $\to l$, Queued $\to q$, Functioning normally $\to n$.

Buffer $\to b$.

Statements: $\sim l \to q$ (1)

$\sim l \leftrightarrow n$ (2)

$\sim q \to b$ (3)

$\sim l \to b$ (4)

$\sim b$ (5)

i) $\sim b$ has to be true $\Rightarrow b = F$.

ii) From (4) we have $l = T$

iii) Similarly from (3) $q = T$.

iv) $n = F$

We thus have a truth assignment for the set of statements.

that ~~makes their conjunction true~~

satisfies the conjunction of set of clauses / statements

$\therefore$ The set of statements is consistent.

c) $((p \vee q) \to (q \vee r))$

This simplifies to $\sim(p \to q) \to r$.

$\therefore$ The 2 expressions are not equivalent.

d)
$$\left.\begin{array}{c} i \to t \\ \underline{\sim s \to \sim t} \\ i \to s \end{array}\right\}$$

Valid.

Use the fact that

$\sim s \to \sim t \equiv t \to s$.

e) Invalid.

2. Prove the following Statements (2 + 3):

   (a) $2^n > n^2$ for sufficiently large n.

   (b) Every integer $n \geq 2$ may be expressed as a product of prime numbers.

Note: n is an arbitrary finite integer.

2a) Base Case: $n = 5$

So, $P(5) = $ True (Trivial)

Assume $P(m) = $ True,

We show $P(m+1) = $ True,

$$2^m > m^2 \quad \text{(Given)}.$$

$$2^m \cdot 2 > m^2 \cdot 2$$

$$2^{m+1} > 2 \cdot m^2 = 2m^2 + 2m + 1 - (2m+1)$$

$$2^{m+1} > (m+1)^2 + m^2 - (2m+1)$$

$$2^{m+1} > (m+1)^2 + \underbrace{(m-1)^2 - 2}_{h}$$

for $m > 5$, $h > 0$

So, $2^{m+1} > (m+1)^2$

Thus, $P(m) \rightarrow P(m+1)$ and $P(5) = $ True.

So, proved $\underline{\forall n > 5}$:

2b) We use strong Induction.

Base case: $n=2$, (trivial)

Consider $P(x) = \text{True} \ \forall \ x \leq m$ (Assumption)

To show, $P(m+1) = \text{True}$.

Now, $m+1$ can have 2 cases:

① It is prime (Trivial).

② It is a product of 2 factors: $n = ab$.

Then, $2 \leq a < n$, $2 \leq b < n$. So, $P(a) = \text{True}$ and $P(b) = \text{True}$.

Thus, <u>proved</u>.

3. The following definitions are used for this problem:

- **Strict Poset :** A strict partially ordered set is a binary relation S on a set X satisfying the conditions:
  - $\forall x, y \in X$, if $(x, y) \in S$, then $(y, x) \notin S$
  - if $(x, y) \in S$ and $(y, z) \in S$, then $(x, z) \in S$
- **Non-Strict Poset :** A non-strict partially ordered set is a binary relation S on a set X satisfying the conditions:
  - $\forall x \in X, (x,x) \in S$.
  - if $(x, y) \in S$ and $(y, x) \in S$, then $x = y$
  - if $(x, y) \in S$ and $(y, z) \in S$, then $(x, z) \in S$
- **Cover :** Let x and y be distinct elements of a set X and let them also be part of the poset R on set X. We say that y covers x if $x <_R y$, but no element z satisfies $x <_R z <_R y$.

True or False, explain your answer (1.5 + 1.5 + 2).

(a) Given a Strict Poset S over the set X, there exists an equivalence relation R on X such that $S' = S \cup R$ is a Non-Strict Poset over X.

(b) Given a Non-Strict Poset S over the set X, there exists an equivalence relation R on X such that $S' = S \backslash R$ is a Strict Poset over X.

(c) Let R be a finite poset on X, and $x, y \in X$. Then $x \leq_R y$ if and only if there exist elements $z_0, ..., z_n \in X$ (for some non-negative integer n) such that $z_0 = x$, $z_n = y$, and z_{i+1} covers z_i for $i = 0, ..., n-1$.

(a) The strict Poset is Irreflective. So, we just need to add all (x,x) $\forall x \in X$ to S in order to get Non-Strict Poset.

$R = \{(x,x) : x \in X\}$. $\longrightarrow$ Equivalence Relation

(b) Again $R = \{(x,x) : x \in X\}$.

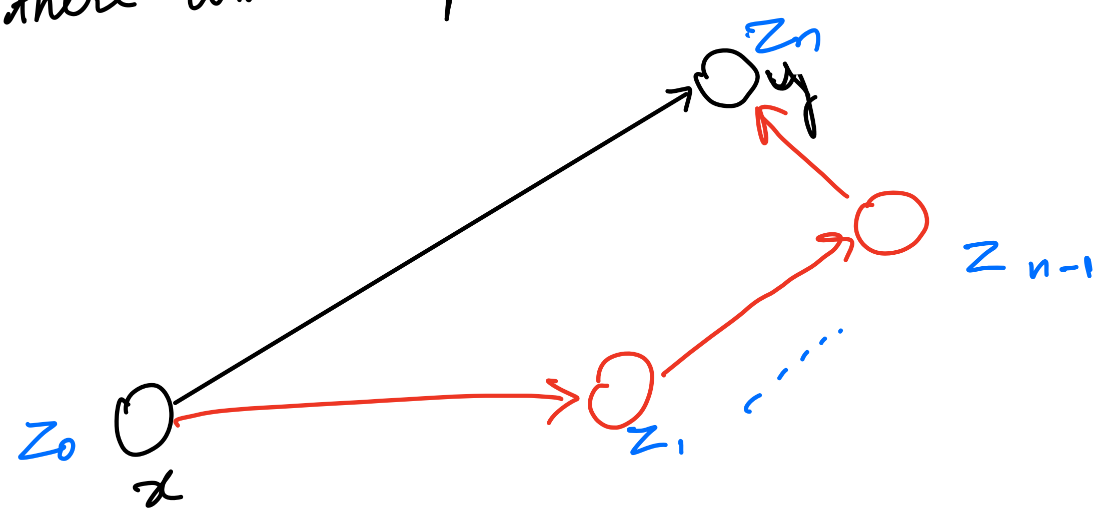
(c) Claim: Every edge in a hasse diagram is a closure.

Proof: By construction of a hasse diagram, we remove an edge if there is an alternate path from source to target. So all remaining edges have no alternate path. This is definition of closure.

Now,
We have to prove both implications:

① $\underbrace{x \leq_R y}_{P_1} \Rightarrow \underbrace{z_0, \dots z_n \in X \text{ exist}}_{P_2}$

Now, if $P_1 = \text{True}$, then x, y are related and during the construction phase of the Hasse diagram, Initially we would draw an edge from x to y. If it is an essential edge, then we keep it. Else we remove it if an alternate path exist. So, in any case, there will be a path from x to y.



From our claim, we know all red edges are closures. Hence proved.

② This implication is trivial and can be shown without a hasse diagram.

4. Consider a game between two players called Kronecker and Cantor. In this game, two parameters, m and n, are introduced, where m and n are positive integers. Kronecker maintains a set $V = \{v_1, v_2, \ldots, v_m\}$ of m binary vectors, each with a length of n. Cantor's objective is to produce a binary vector u, also of length n, that differs from each v_i, or to declare that no such vector exists. To accomplish this, Cantor is permitted to ask queries, with each query taking the form:

"What is the value of bit j in vector i ?" where $1 \leq j \leq n$ and $1 \leq i \leq m$.

Kronecker is responsible for responding to each query posed by Cantor. The main goal of Cantor is to minimize the number of queries required to produce u.

Note: Cantor presents his queries to Kronecker sequentially and may decide on the next query based on Kronecker's responses to the previous ones.

You are going to help Cantor. Write down the best strategy you can come up with and the number of queries for these strategies for each of the given cases (1.5 + 1 + 2.5).

(a) $m = n$.

(b) Briefly prove that it is impossible to guarantee a vector u different from all v_i, if the number of queries is less than m for part (a).

(c) $m = n + 1$. **Hint:** Try to use part (a).

(a) direct application of Cantor's diagonalisation.

$u[i] = 1 - v_i[i]$ where $x[i]$ is i^{th} element of binary vector x.

So, ~~total~~ m queries.

(B) If the number of queries is less than m, and we try to get information about a different v_i in each query so that u is different from v_i, there will be at least 1 vector v_j which we would not be able to query. Since we know nothing about v_j, there is a non-zero prob. that $u = v_j$. thus no. of queries $\geq m$.

(c) We decide the first element of u, using v_1, v_2 v_3. Note that at least 2 of v_1, v_2, v_3 must have the same first bit $b \in \{0,1\}$. Now choose $u[1] = b$. Then u is different from 2 of v_1, v_2, v_3. Now we have $n-1$ bits to decide and $m-2$ vectors to be different from.

$$m - 2 = n - 1.$$

So, our proble has now been converted to $m' = n-1$ and $n' = n-1$, as $m' = n'$, we apply the solution of (a).