SOLUTIONS OF ASSIGNMENT 1

In this assignment, you are required to work by yourself to successfully complete two activities, each encompassing four distinct parts, as detailed below.

Note that in this assignment

- the symbol $\overline{9b}$ denotes a two-digit natural number,
- the symbol $\overline{12a}$ represents a three-digit natural number.

Specifically, suppose $a = 3$, $\overline{12a}$ is equivalent to 123.

Let $a < b$ be the largest digits in your student ID.

Activity 1

For each student, replace a and b with specific numerical values to obtain the ultimate result.

Part I. Stick in mind that $a < b$ represents the largest digits in your ID.

1. Let A and B be two non-empty finite sets. Assume that cardinalities of the sets A, B, and $A \cap B$ are $\overline{9b}$, $\overline{2a}$ and $a + b$, respectively. Determine the cardinality of the set $A \cup B$.

   Hint. We have

$$\begin{aligned} |A \cup B| &= |A| + |B| - |A \cap B| \\ &= \overline{9b} + \overline{2a} - (a + b) \\ &= (90 + b) + (20 + b) - (a + b) = 110. \end{aligned}$$

2. Suppose $|A - B| = \overline{3a}$, $|A \cup B| = \overline{11b}$ and $|A \cap B| = \overline{1a}$. Determine $|B|$.

   Hint. We have

$$|A \cup B| = |A - B| + |B - A| + |A \cap B|$$

   which implies

$$\overline{11b} = \overline{3a} + |B - A| + \overline{1a}.$$

   It follows that

$$|B - A| = \overline{11b} - \overline{3a} - \overline{1a} = (110 + b) - (30 + a) - (10 + a) = 70 + b - 2a.$$

In addition,

$$\begin{aligned}|B| &= |B - A| + |A \cap B| \\ &= 70 + b - 2a + \overline{1a} \\ &= 80 + b - a.\end{aligned}$$

3. At a local market, there are $\overline{35b}$ customers. Suppose $\overline{11a}$ have purchased fruits, $\overline{9b}$ have purchased vegetables, $\overline{8a}$ have purchased bakery items, $\overline{4b}$ have purchased both fruits and vegetables, $\overline{3b}$ have purchased both vegetables and bakery items, $\overline{2a}$ have purchased both fruits and bakery items, and $\overline{1a}$ have purchased all three categories. How many customers have not purchased anything?

   Hint. To find the number of customers who have not purchased anything, we use the formula

$$\begin{aligned}Total =& Fruits + Vegetables + Bakery \\ & - (Fruits\ \&\ Vegetables + Vegetables\ \&\ Bakery + Fruits\ \&\ Bakery) \\ & + All\ Three + Neither\end{aligned}$$

   where

   - $Total$ is the total number of customers $\overline{35b}$,
   - $Fruits$ is the number of customers who purchased fruits $\overline{11a}$,
   - $Vegetables$ is the number of customers who purchased vegetables $\overline{9b}$,
   - $Bakery$ is the number of customers who purchased bakery items $\overline{8a}$,
   - $Fruits\ \&\ Vegetables$ is the number of customers who purchased both fruits and vegetables $\overline{4b}$,
   - $Vegetables\ \&\ Bakery$ is the number of customers who purchased both vegetables and bakery items $\overline{3b}$,
   - $Fruits\ \&\ Bakery$ is the number of customers who purchased both fruits and bakery items $\overline{2a}$,
   - $All\ Three$ is the number of customers who purchased all three categories $\overline{1a}$,
   - $Neither$ is the number of customers who have not purchased anything (what we are trying to find).

It follows that

$$\overline{35b} = \overline{11b} + \overline{9b} + \overline{8a} - (\overline{4b} + \overline{3b} + \overline{2a}) + \overline{1a} + Neither$$

which implies

$$Neither = 150 - 2a + 2b.$$

We conclude that there are $150 - 2a + 2b$ customers have not purchased anything.

Part II. Keep in mind that $a < b$ represents the largest digits in your ID.

1. List the bag of prime factors for each of the provided numbers.

   a) $\overline{1a2}$.

   b) $\overline{2b0}$.

   Hint.Depending on each value of a and b, we proceed with two steps as follows:

   - First, prime factorization for each number provided.
   - Second, list the factors as bags.

   For instance, in the case $a = 1$, we have

   - First, $\overline{1a2} = 112 = 2^4 \times 7$.
   - Second, the bag is $\{\!|2 : 4, 7 : 1|\!\}$.

   Similarly applicable for the specific value of b.

2. Find the cardinalities of

   a) each of the aforementioned bags.

   b) the intersection of the aforementioned bags.

   c) the union of the aforementioned bags.

   d) the difference of the aforementioned bags.

   Hint. Applying the formulas in section 3 Bags Manipulation Functions in Lecture 3.

Part III. Bear in mind that $a < b$ represents the largest digits in your ID.

1. Ascertain whether the given functions are invertible. If they are, identify the rule for the inverse function f^{-1}.

a) $f: \mathbb{R} \to \mathbb{R}$ with $f(x) = bx + a$.

b) $f: [-b, +\infty) \to [0, +\infty)$ with $f(x) = \sqrt{x+b}$.

Hint.

a) In this case, $b \neq 0$, the function f has the inverse $f^{-1}(y) = \dfrac{y-a}{b}$.

b) The function f has the inverse $f^{-1}(y) = y^2 - b$.

2. Let $f, g: \mathbb{R} \to \mathbb{R}$ be defined as

$$f(x) = \begin{cases} 2x + a & \text{if } x < 0 \\ x^3 + b & \text{if } x \geq 0 \end{cases} \quad \text{and} \quad g(x) = bx - a.$$

Find $g \circ f$.

Hint. We have

$$g \circ f(x) = g(f(x)) = \begin{cases} b(2x + a) - a & \text{if } x < 0 \\ b(x^3 + b) - a & \text{if } x \geq 0 \end{cases} = \begin{cases} 2bx + ba - a & \text{if } x < 0 \\ bx^3 + b^2 - a & \text{if } x \geq 0. \end{cases}$$

Part IV. Show that if A, B, and C are sets, then $\overline{A \cup B \cup C} = \overline{A} \cap \overline{B} \cap \overline{C}$.

1. Demonstrate that each side is a subset of the other side.

Hint.

◇ *Forward inclusion*: $\overline{A \cup B \cup C} \subseteq \overline{A} \cap \overline{B} \cap \overline{C}$.

- Let $x \in \overline{A \cup B \cup C}$.
- This implies $x \notin A \cup B \cup C$.
- By De Morgan' Law, this is equivalent to x not being in A, B and C individually.
- Thus, $x \in \overline{A}$, $x \in \overline{B}$ and $x \in \overline{C}$.
- Consequently, we obtain $x \in \overline{A} \cap \overline{B} \cap \overline{C}$.

◇ *Reverse Inclusion*: $\overline{A} \cap \overline{B} \cap \overline{C} \subseteq \overline{A \cup B \cup C}$.

- Let $y \in \overline{A} \cap \overline{B} \cap \overline{C}$.
- This implies y not in A, B and C individually.
- By De Morgan's Law, this is equivalent to y not being in $\overline{A \cup B \cup C}$.
- Hence, $y \in \overline{A \cup B \cup C}$.

2. Verify the equality using a membership table.

Hint.

A	B	C	$A \cup B \cup C$	$\overline{A \cup b \cup C}$	$\overline{A} \cap \overline{B} \cap \overline{C}$
0	0	0	0	1	1
0	0	1	1	0	0
0	1	0	1	0	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	0	0
1	1	0	1	0	0
1	1	1	1	0	0

Activity 2

For each student, replace a and b with specific numerical values to obtain the ultimate result.

Part I. Discuss two notable instances of binary trees, providing both quantitative and qualitative analyses.

Hint.

1. Binary Tree Data Structure:
   - **Definition**: A binary tree is a hierarchical data structure composed of nodes, each having at most two children, referred to as the left child and the right child.
   - **Root**: The topmost node in a binary tree is called the root.
   - **Nodes**: Each node in a binary tree contains data and references (or links) to its left and right children.
   - **Child Nodes**: Nodes that are connected below another node are its children. A node with no children is a leaf node.
   - **Depth**: The level of a node is its depth in the tree, with the root having depth 0.
   - **Height**: The height of a binary tree is the length of the longest path from the root to a leaf.
   - **Applications**: Binary trees are widely used in computer science for tasks like searching, sorting, expression parsing, and hierarchical representation of data.

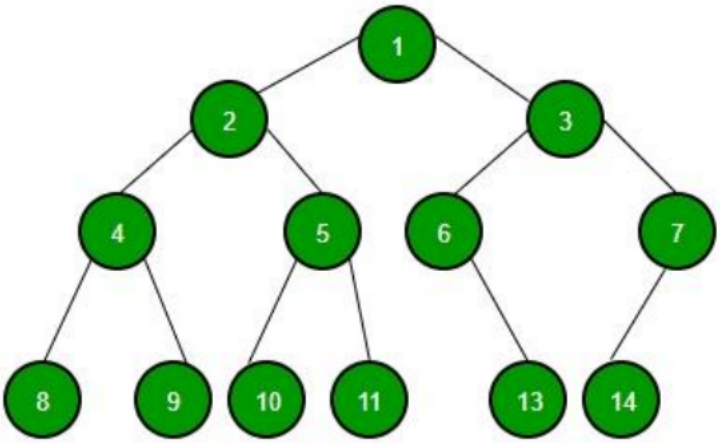


Figure: Binary Tree.

2. The following are prevalent types of Binary Trees:

   - **Full binary tree**: A full binary tree is a type of binary tree in which every node has either 0 or 2 children.
     - **Node Count**: The total number of nodes in a full binary tree with height h is $2^{h+1} - 1$.
     - **Leaf Nodes**: All leaf nodes in a full binary tree are at the same level or have the same depth.
     - **Structure**: The structure of a full binary tree ensures that every level, except possibly the last, is completely filled, and all nodes are as left as possible.
     - **Examples**: *Perfect binary trees* are a specific type of full binary tree where all leaf nodes are at the same level.

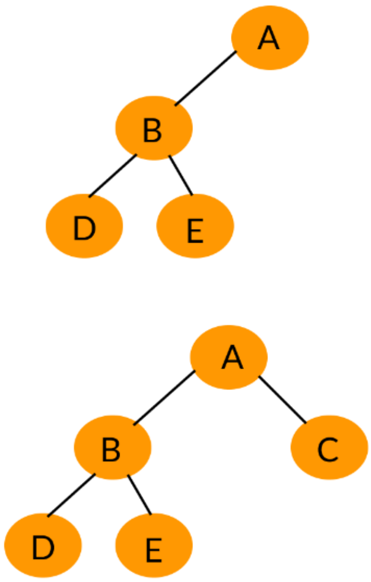


Figure: Full Binary Tree.

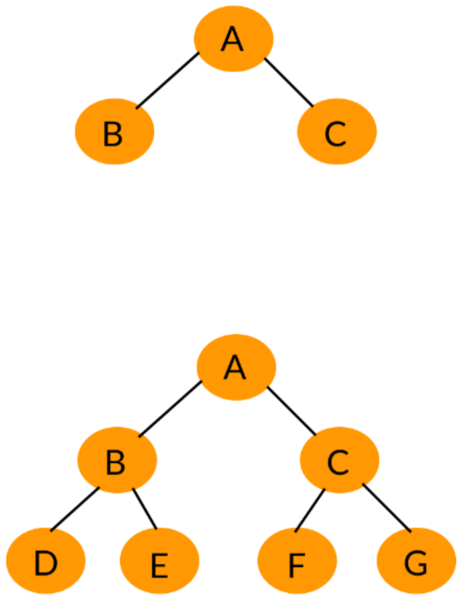


Figure: Perfect Binary Tree.

- **Complete binary tree**: A complete binary tree is a binary tree in which every level, except possibly the last, is completely filled, and all nodes are as left as possible.
  - **Node Count**: If a complete binary tree has height h, it contains between 2^h and $2^{h+1} - 1$ nodes.
  - **Leaf Nodes**: In a complete binary tree, leaf nodes are only present at the last level, and they are positioned from left to right.
  - **Structure**: The structure of a complete binary tree allows for efficient storage using an array representation.

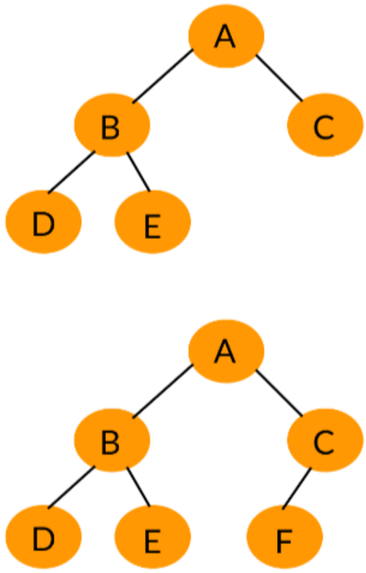


Figure: Complete Binary Tree.

- **Balanced binary tree**: A balanced binary tree, also known as a height-balanced or AVL tree, is a binary tree in which the heights of the two child subtrees of any node differ by at most one.

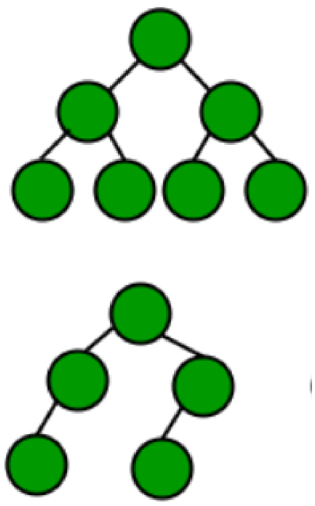


Figure: Balanced Binary Trees of Height 3.

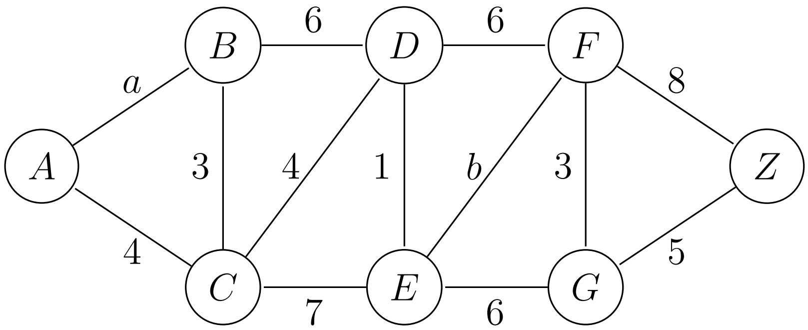
Part II. Stick in mind that $a < b$ represents the largest digits in your ID.

1. State Dijkstra's Algorithm in an undirected graph.

   Hint. We proceed as the following steps:

   - Initialize the distance to the source node as 0 and all other distances as infinity $+\infty$.
   - Mark all nodes as unvisited.
   - Repeat until all nodes are visited:
     - Select the unvisited node with the smallest tentative distance.
     - For each neighboring node not yet visited: Update the tentative distance if the sum of the current distance and edge weight is smaller.
   - The final distances represent the shortest paths from the source to all other nodes.

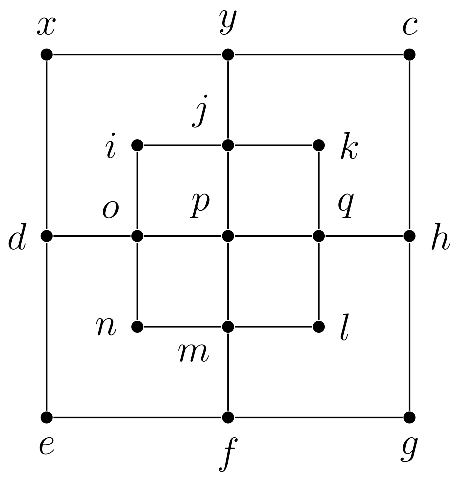
2. Apply Dijkstra's algorithm to determine the shortest path length between vertices A and Z in the provided weighted graph.



   Hint. Utilize Dijkstra's Algorithm as described above for each particular scenario of both a and b. Additionally, refer to the specific instructions in Section 1 of Lecture 8.

Part III. Does the following graph have a Hamilton/ Euler path? If so, find such a path. If it does not, give an argument to show why no such path exists.

Hint. No Hamilton path exists. There are eight vertices of degree 2, and only two of them can be end vertices of a path. For each of the other six, their two incident edges must be in the path. It is not hard to see that if there is to be a Hamilton path, exactly one of the inside corner vertices must be an end, and that this is impossible. Similar to the Euler path.



Part IV. Construct a proof of the Five Color Theorem. Refer to the following document to complete the proof: Chapter 4 The Five-Color Theorem.

SOLUTIONS OF ASSIGNMENT 2

In this assignment, you are required to work in a group to successfully complete two activities, each encompassing four distinct parts, as detailed below.

Activity 1

Part I. Discuss the utilization of Boolean Algebra in addressing binary problems encountered in two diverse real-world domains and the practical applications therein.

Hint. Boolean Algebra is crucial in solving binary problems in diverse real-world domains.

- In computer science, it plays a fundamental role in digital circuit design, where binary logic gates use Boolean operations for processing and controlling information flow.
- Additionally, in computer programming and software development, Boolean algebra is employed in decision-making processes, conditionals, and algorithms, contributing to efficient problem-solving.

Overall, Boolean Algebra provides a foundational framework for addressing binary challenges in both hardware and software domains, enabling practical solutions in various real-world applications.

◇ **Representing numbers in binary** is a fundamental concept in computer science and digital systems. Here are some key points about representing numbers in binary:

- *Binary System*: The binary system is a base-2 numeral system that uses two digits, 0 and 1. Each digit in a binary number is called a *bit* (binary digit).
- *Binary Digits (Bits)*: A bit represents the smallest unit of data in computing and can have a value of 0 or 1.
- *Conversion from Decimal to Binary*: To convert a decimal number to binary, repeatedly divide the decimal number by 2 and note the remainders from bottom to top.
  <u>Example</u>: The binary representation of 27 is 11011.

- *Conversion from Binary to Decimal*: To convert a binary number to decimal, multiply each bit by 2 raised to the power of its position and sum the results.

  Example: The decimal equivalent of 101101 in binary is 53, namely

  $$1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 53.$$

- *Application in Computers*: Binary representation is fundamental in computer architecture and digital electronics. All data and instructions in a computer's memory are represented in binary.
- *Hexadecimal Notation*: Hexadecimal (base-16) is often used to represent binary values more compactly. Each hexadecimal digit corresponds to four binary digits.

◇ **Using logic gates** with binary is a fundamental concept in digital electronics and computer science. Here are several key points related to this topic:

- *Logic Gates*:
  * Logic gates are electronic devices that perform logical operations on one or more binary inputs to produce a binary output.
  * Common logic gates include AND, OR, NOT, XOR (exclusive OR), NAND (NOT AND), and NOR (NOT OR).
- *Binary Input and Output*:
  * Logic gates operate with binary input signals, where each input can be either 0 or 1.
  * The output of a logic gate is also binary, resulting in either 0 or 1.
- *AND Gate*:
  * The output of an AND gate is 1 only if all of its inputs are 1; otherwise, the output is 0.
  * Boolean expression $C = A \cdot B$ where C is the output.
- *OR Gate*:
  * The output of an OR gate is 1 if at least one of its inputs is 1; otherwise, the output is 0.
  * Boolean expression $C = A + B$ where C is the output.

- *NOT Gate*:
  - The output of a NOT gate is the opposite (complement) of its input.
  - Boolean expression $B = \overline{A}$ where B is the output.

Part II.

1. Develop the truth table and derive the corresponding Boolean equation for each of the following scenarios.

   a) "In a secure facility, if the access card is swiped OR the correct PIN is entered AND the security system is NOT in maintenance mode, then the door should unlock."

   Hint. Let's break down the given scenario into its components and create a truth table

   - A represents "Access card is swiped."
   - P represents "Correct PIN is entered."
   - M represents "Security system is in maintenance mode."
   - D represents "Door unlocks."

   The statement can be expressed as

$$(A \text{ OR } P) \text{ AND } (\text{NOT } M) \ \rightarrow \ D.$$

   The truth table is

A	P	M	$(A$ OR $P)$ AND NOT M	D
0	0	0	0	0
0	0	1	0	0
0	1	0	1	1
0	1	1	0	0
1	0	0	1	1
1	0	1	0	0
1	1	0	1	1
1	1	1	0	0

   The Boolean equation for the scenario is

$$D = A \cdot \overline{M} + P \cdot \overline{M}.$$

   This equation represents the logical conditions for the door to unlock based on the given scenario.

b) "For a computer to successfully log in, either a valid username and password combination must be entered OR a security token must be provided, but not both."

Hint. Let's break down the given scenario into its components and create a truth table

- U represents "Valid username and password combination is entered."
- P represents "Security token is provided."
- L represents "Computer successfully logs in."

The statement can be expressed as

$$[U \text{ AND } (\text{NOT } P)] \text{ OR } [(\text{NOT } U) \text{ AND } P] \; \rightarrow \; L.$$

The truth table is

U	P	$[U$ AND (NOT P)] OR [(NOT U) AND $P]$	L
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0

The Boolean equation corresponding to the given scenario is

$$L = (U \cdot \overline{P}) + (\overline{U} \cdot P).$$

This equation represents the logical conditions for the computer to successfully log in based on the given scenario.

2. Generate a truth table for the provided Boolean expression.

$$xyz + x\bar{y}\bar{z} + \bar{x}y\bar{z} + \bar{x}\bar{y}z.$$

Hint.

x	y	z	xyz	$x\bar{y}\bar{z}$	$\bar{x}y\bar{z}$	$\bar{x}\bar{y}z$	$xyz + x\bar{y}\bar{z} + \bar{x}y\bar{z} + \bar{x}\bar{y}z$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	1	0	0	1	0	1
1	0	0	0	0	0	0	0
1	0	1	0	0	0	1	1
1	1	0	0	1	0	0	1
1	1	1	1	0	0	0	1

Part III. Simplify the following Boolean expressions.

1. $x(x+y)+y(y+z)+z(z+x)$.

   Hint. $x^2+xy+y^2+yz+z^2+zx$.

2. $(x+\bar{y})(y+z)+(x+y)(z+\bar{x})$.

   Hint. $x+yz$.

3. $(x+y)(xz+x\bar{z})+zx+x$.

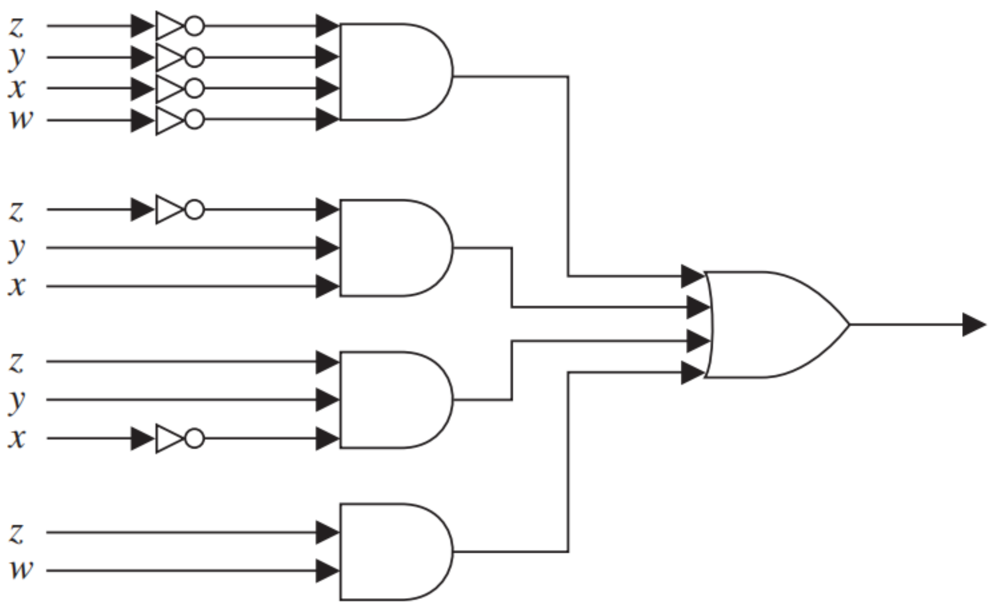
   Hint. 1.

4. $\bar{x}(x+y)+(x+y)(x+\bar{y})$.

   Hint. $x+y$.

Part IV. Build a circuit using OR gates, AND gates, and inverters that produces an output of 1 if a decimal digit, encoded using a binary coded decimal expansion, is divisible by 3, and an output of 0 otherwise. Identify the representation of the Boolean function that has the output value.

Hint.



Activity 2

Part I.

1. Explain the attributes of various binary operations executed within a common set.

   Hint.

   - ◇ **Definition**:
     - A binary operation on a set S is a rule that assigns to each ordered pair of elements in S a unique element in S.
     - Denoted as $a \circ b$ or $a * b$, where $a, b \in S$.
   - ◇ **Closure**:
     - A binary operation is closed if, for any $a, b \in S$, the result $a \circ b$ is also in S.
     - Closure is a fundamental property of binary operations.
   - ◇ **Associativity**:
     - An operation is associative if $(a \circ b) \circ c = a \circ (b \circ c)$ for all $a, b, c \in S$.
     - The grouping of elements does not affect the result.
   - ◇ **Commutativity**:
     - An operation is commutative if $a \circ b = b \circ a$ for all $a, b \in S$.
     - Not all operations are commutative (e.g., subtraction).
   - ◇ **Identity Element**:
     - An element $e \in S$ is an identity element if $a \circ e = a$ and $e \circ a = a$ for all $a \in S$.
     - Not all sets have identity elements.
   - ◇ **Inverse Element**:
     - An element b is the inverse of a if $a \circ b = e$ and $b \circ a = e$, where e is the identity element.
     - Not all elements have inverses.
   - ◇ **Distributivity**:
     - For sets with two binary operations ($+$ and $\cdot$), if $a \cdot (b+c) = a \cdot b + a \cdot c$ and $(a+b) \cdot c = a \cdot c + b \cdot c$, the operations are distributive.

- ◇ **Idempotence**:
  - An element a is idempotent if $a \circ a = a$.
  - Some operations, like the logical AND or OR, are idempotent.
- ◇ **Binary Operations in Mathematics**: Common examples include addition and multiplication in arithmetic, union and intersection in set theory, and logical operations in Boolean algebra.
- ◇ **Semigroup, Monoid, Group,...**: These are algebraic structures defined based on properties of binary operations.

2. Check whether the operations applied to pertinent sets qualify as binary operations.

   a) Subtraction on set of natural numbers.

$$\begin{aligned} - :\ & \mathbb{N} \times \mathbb{N} \to \mathbb{N} \\ & (x, y) \mapsto x - y. \end{aligned}$$

   Hint. As properties of binary operation, subtraction is not a binary operation on $\mathbb{N}$.

   b) Exponential operation on set integers.

$$\begin{aligned} \hat{}:\ & \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z} \\ & (x, y) \mapsto x^y. \end{aligned}$$

   Hint. As properties of binary operation, exponential operation is not a binary operation on $\mathbb{Z}$.

Part II.

1. Construct the operation tables for group G with orders 1, 2, 3, and 4, utilizing the elements a, b, c, and e as the identity element in a suitable manner.

   Hint. In group theory, the order of a group is the number of elements in the group. We now construct operation tables for groups of orders $1, 2, 3$, and 4, using the elements a, b, c and e as the identity element.

   Remember, for a group, each element must appear *exactly once* in each row and each column of the operation table, and every combination of two elements must produce another element in the group.

- **Order 1**: For a group of order 1, there is only one element (say e), and the operation table is as follows

$\circ$	e
e	e

- **Order 2**: For a group of order 2, we can use elements e and a. The table is

$\circ$	e	a
e	e	a
a	a	e

- **Order 3**: For a group of order 3, we can use elements e, a and b. The table is

$\circ$	e	a	b
e	e	a	b
a	a	b	e
b	b	e	a

- **Order 4**: For a group of order 4, we can use elements e, a, b and c. The table is

$\circ$	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

The tables are constructed such that the group properties are satisfied, including closure, associativity, identity element, and inverses.

2. State the Lagrange's theorem of group theory. Using this theorem to discuss whether a group K with order 4 can be a subgroup of a group G with order 9 or not. Provide a clear exposition of the reasons.

Hint.

- **Lagrange's Theorem**: For any finite group G and any subgroup H of G the order of H divides the order of G.
- **Application**: Based on Lagrange's Theorem, a group with order 4 cannot be a subgroup of a group with order 9. There is no subgroup of order 4 in a group of order 9 that satisfies Lagrange's condition.

Part III.

1. Check whether the set $S = \mathbb{R} - \{-1\}$ is a group under the binary operation $*$ defined as $a * b = a + b + ab$ for any $a, b \in S$.

   Hint. The set $S = \mathbb{R} - \{-1\}$ is not a group under the binary operation $*$ defined as $a * b = a + b + ab$ because it does not satisfy the group axioms. Specifically, it lacks closure since the product $a + b + ab$ can be equal to -1, which is excluded from S. Additionally, there is no identity element or inverses within the set, violating the requirements for a group.

2. Express the connection between the order of a group and the quantity of binary operations that can be defined on that set. Apply to answer the question "What is the total number of binary operations that can be defined on a set containing 3 elements?"

   Hint.

   - The number of binary operations that can be defined on a set is related to the order of the group. For a set with n elements, there are n^{n^2} possible binary operations.
   - For a set containing 3 elements, there are 3^{3^2} possible binary operations.

Part IV. Prepare a fifteen minutes presentation that explains an application of group theory in computer sciences.

Hint. Here are several key terms relevant to the application of group theory in computer science, which could aid students in selecting a topic for their presentations: Error Detection and Correction, Cryptography, Computer Graphics and Animation, Database Query Optimization, Network Security, Machine Learning, Computer Algebra Systems, Algorithm Design, ...