Bihar Board Class 12 Computer Science (ADDL.) Set B 2025 Question Paper with Solutions

Time Allowed :3 Hours 15 Minutes	Maximum Marks :70	Total Questions :96

General Instructions

Read the following instructions very carefully and strictly follow them:

1. The test is of 3 hours 15 Minutes duration.
2. The question paper consists of 96 questions.
3. In Section - B, there are 20 short answer type questions, each carrying 2 marks, out of which any 10 questions are to be answered. Apart from these, there are 6 long answer type questions, each carrying 5 marks, out of which any 3 questions are to be answered.
4. Minimum 30% marks in each subject (30 out of 100 for theory, adjusted for practicals where applicable).
5. Use of any electronic appliances is strictly prohibited.

Section - A

1. Boolean algebra is basically based on

(A) Truth
(B) Logic
(C) Symbol
(D) Numbers

Correct Answer: (B) Logic

Solution:

Step 1: Understanding the Concept:
Boolean algebra is a branch of algebra in which the values of the variables are the truth values *true* and *false*, usually denoted as 1 and 0, respectively. It was introduced by George Boole in the 19th century.

Step 2: Detailed Explanation:
The fundamental purpose of Boolean algebra is to analyze and simplify logical statements and circuits. It operates on logical propositions and uses operators like AND, OR, and NOT, which are direct representations of logical operations.

- **(A) Truth:** Truth (and Falsity) are the *values* used within Boolean algebra, not the foundation it is based on.

- **(B) Logic:** Boolean algebra is the mathematical formalization of logic. It provides a system for manipulating logical statements. This is the correct basis.

- **(C) Symbol:** Symbols (like variables A, B and operators $+$, $\cdot$) are used to *represent* logical concepts, but the underlying principle is logic itself.

- **(D) Numbers:** While we often use numbers (0 and 1) to represent false and true, the algebra is about logical relationships, not numerical calculations in the traditional sense.

Step 3: Final Answer:
Therefore, Boolean algebra is fundamentally based on logic.

Quick Tip

Remember that Boolean algebra is the backbone of digital electronics and computer science. Every decision a computer makes is boiled down to the simple logical principles defined by Boolean algebra.

---

2. The output of OR gate is 1 when

(A) If both inputs are zero
(B) If either or both inputs are 1
(C) Only if both inputs are 1
(D) If either input is zero

Correct Answer: (B) If either or both inputs are 1

Solution:

Step 1: Understanding the Concept:
An OR gate is a fundamental digital logic gate that implements logical disjunction. Its output is high (1 or true) if one or more of its inputs are high (1 or true).

Step 2: Key Formula or Approach:
The truth table for a 2-input OR gate is as follows:

Input A	Input B	Output (A OR B)
0	0	0
0	1	1
1	0	1
1	1	1

Step 3: Detailed Explanation:
By examining the truth table, we can analyze the given options:

- **(A) If both inputs are zero:** The output is 0. This is incorrect.
- **(B) If either or both inputs are 1:** This covers the cases (0,1), (1,0), and (1,1). In all these cases, the output is 1. This is the correct description.
- **(C) Only if both inputs are 1:** This describes the AND gate, not the OR gate. This is incorrect.
- **(D) If either input is zero:** This is not a sufficient condition. In the case (0,0), the output is 0. In the cases (0,1) and (1,0), the output is 1. The statement is ambiguous and not fully correct.

Step 4: Final Answer:
The output of an OR gate is 1 when at least one of its inputs is 1, which is best described as "if either or both inputs are 1".

Quick Tip

A simple way to remember the OR gate's function is to think of it as "any input is enough". If any input is 1, the output will be 1. The only way to get a 0 is if all inputs are 0.

---

3. Which of the following is a functionally complete set of gates ?
(i) NAND (ii) NOT

(A) I but not II
(B) II but not I
(C) Both I and II
(D) None of these

Correct Answer: (A) I but not II

Solution:

Step 1: Understanding the Concept:
A "functionally complete" set of logic gates (also known as a universal set) is a set of gates from which any possible Boolean function can be realized. This means we must be able to create

the three basic logic functions: AND, OR, and NOT, using only gates from that set.

Step 2: Detailed Explanation:
Let's analyze the given gates:

- **(i) NAND gate:** The NAND gate is a universal gate. We can implement AND, OR, and NOT using only NAND gates.
  - **NOT:** NOT A = A NAND A
  - **AND:** A AND B = (A NAND B) NAND (A NAND B)
  - **OR:** A OR B = (A NAND A) NAND (B NAND B)
  Since all basic functions can be derived, the set {NAND} is functionally complete.

- **(ii) NOT gate:** The NOT gate (inverter) can only perform inversion. It cannot create the AND or OR functions. Therefore, the set {NOT} is not functionally complete.

Step 3: Final Answer:
Based on the analysis, the NAND gate (i) forms a functionally complete set, but the NOT gate (ii) does not. Thus, the correct option is "I but not II".

Quick Tip

Memorize the two universal gates: NAND and NOR. Either of these gates, by itself, constitutes a functionally complete set. This is a very common topic in digital logic questions.

---

4. Which of the following will not return a value ?

(A) Null
(B) Void
(C) Empty
(D) Free

Correct Answer: (B) Void

Solution:

Step 1: Understanding the Concept:
In many programming languages (like C, C++, Java, C#), functions are designed to perform a task and often return a result or value. The "return type" of a function specifies the data type of the value that the function sends back to the caller. When a function is not supposed to return any value, a special keyword is used.

Step 2: Detailed Explanation:
Let's analyze the options:

- **(A) Null:** 'Null' is a special *value* that typically represents the absence of a reference to an object. A function can return 'null', so it does not mean "not returning a value." For example, a function that searches for an object might return 'null' if the object is not found.

- **(B) Void:** The 'void' keyword is specifically used as a return type to indicate that a function does not return any value. Its purpose is to explicitly state that no value is sent back. For example: 'void printMessage() { ... }'.

- **(C) Empty:** "Empty" is a state or a property, not a return type keyword. For instance, a function can return an "empty" string ('""') or an "empty" list, but these are still valid values of their respective data types (string and list).

- **(D) Free:** 'free()' is a function in C used for deallocating memory. It itself is a function that has a 'void' return type, but "Free" is not a keyword to indicate no return value.

Step 3: Final Answer:
The correct term to specify that a function will not return a value is 'void'.

Quick Tip

Distinguish carefully between 'void' and 'null'. 'void' means a function returns absolutely nothing. 'null' is an actual value that a function can return, representing "no object".

---

5. Terabyte (TB) is the unit of measurement for

(A) Light wave
(B) Speed of aircraft
(C) Computer memory
(D) Bandwidth

Correct Answer: (C) Computer memory

Solution:

Step 1: Understanding the Concept:
In computing, data storage capacity is measured in units like bytes, kilobytes, megabytes, gigabytes, and terabytes. These units quantify the amount of digital information that can be

stored in a device.

Step 2: Detailed Explanation:

- **Terabyte (TB):** A terabyte is a unit of digital information storage. Conventionally, 1 TB is equal to 1,024 gigabytes (GB) or approximately 1 trillion bytes. It is commonly used to describe the capacity of storage devices like hard disk drives (HDDs), solid-state drives (SSDs), and Random Access Memory (RAM). Therefore, it is a unit for **computer memory** and storage.

- **(A) Light wave:** Light waves are measured by their wavelength (e.g., in nanometers) or frequency (e.g., in Hertz).

- **(B) Speed of aircraft:** Speed is measured in units like kilometers per hour (km/h) or miles per hour (mph).

- **(D) Bandwidth:** Network bandwidth, the rate of data transfer, is typically measured in bits per second (bps) or its multiples like megabits per second (Mbps) or gigabits per second (Gbps).

Step 3: Final Answer:
A terabyte (TB) is a unit of measurement for computer memory and data storage capacity.

Quick Tip

Remember the hierarchy of data storage units: Kilobyte (KB), Megabyte (MB), Gigabyte (GB), Terabyte (TB), Petabyte (PB). Each is roughly 1000 (or exactly 1024) times larger than the previous one.

---

6. How are the constant declared ?

(A) Const keyword
(B) # define preprocessor
(C) Both (A) and (B)
(D) S define

Correct Answer: (C) Both (A) and (B)

Solution:

Step 1: Understanding the Concept:
A constant is a value in a program that does not change during execution. Languages like C and C++ provide multiple ways to define such constants.

Step 2: Detailed Explanation:
Let's examine the common methods:

- **(A) 'const' keyword:** This method declares a variable as read-only. The compiler enforces that its value cannot be changed. This approach is type-safe and respects variable scope.
  Example: 'const float PI = 3.14159;'

- **(B) 'define' preprocessor directive:** This is a C/C++ preprocessor macro. Before compilation, the preprocessor scans the code and replaces every occurrence of the macro's name with its defined value. It is essentially a text substitution and is not type-safe.
  Example: 'define PI 3.14159'

Since both 'const' and 'define' are valid and widely used methods for creating constants in C and C++, the correct answer is that both can be used.

Step 3: Final Answer:
Constants can be declared using both the 'const' keyword and the 'define' preprocessor directive.

Quick Tip

In modern C++, it is strongly recommended to use 'const' (or 'constexpr') over 'define' for declaring constants. 'const' is type-safe and respects scope, which leads to more robust and less error-prone code.

---

7. Which one of the following is used with cout as an operator ?

(A) <<
(B) >>
(C) >
(D) <

Correct Answer: (A) <<

Solution:

Step 1: Understanding the Concept:
In C++, 'cout' is an object of the 'ostream' class, representing the standard output stream

(usually the console). To send data to this stream, a specific operator is used.

Step 2: Detailed Explanation:
The operators related to C++ streams are:

- **(A) ‘<<‘ (Stream Insertion Operator):** This operator is used to "insert" data into an output stream. With ‘cout‘, it sends data to the console. For example: ‘cout << "Hello, World!";‘

- **(B) ‘>>‘ (Stream Extraction Operator):** This operator is used to "extract" data from an input stream, such as ‘cin‘ (the standard input stream). For example: ‘cin >> myVariable;‘

- **(C) ‘>‘ and (D) ‘<‘ (Relational Operators):** These are the "greater than" and "less than" operators, used for comparing values in conditional statements, not for stream operations.

Step 3: Final Answer:
The operator used with ‘cout‘ to display output is the stream insertion operator, ‘<<‘.

Quick Tip

Think of the direction of the operator as the direction of data flow. For ‘cout ¡¡ data;‘, the data flows from the variable to ‘cout‘. For ‘cin ¿¿ variable;‘, the data flows from ‘cin‘ to the variable.

---

8. If all devices are connected to a central hub then this topology is called

(A) Bus topology
(B) Ring topology
(C) Star topology
(D) Tree topology

Correct Answer: (C) Star topology

Solution:

Step 1: Understanding the Concept:
Network topology refers to the physical or logical arrangement of nodes (like computers and printers) and connections within a network.

Step 2: Detailed Explanation:
Let's define the given topologies:

- **(A) Bus Topology:** All devices are connected to a single central cable, called the bus or backbone.

- **(B) Ring Topology:** Each device is connected to exactly two other devices, forming a single continuous pathway for signals through each node - a ring.

- **(C) Star Topology:** All devices are connected to a central device, such as a hub or a switch. Each device has a dedicated point-to-point link to the central hub. This exactly matches the description in the question.

- **(D) Tree Topology:** This is a hybrid topology that combines characteristics of bus and star topologies. It consists of groups of star-configured workstations connected to a linear bus backbone cable. While it uses hubs, the fundamental structure described in the question is a single star.

Step 3: Final Answer:
A network where all devices connect to a central hub is called a star topology.

Quick Tip

Visualize the shapes to remember topologies: a single line is a **Bus**, a circle is a **Ring**, and a central point with spokes is a **Star**. The main advantage of a star topology is that if one link fails, only that single node is isolated.

---

9. Which of the following is most suitable for a menu-driven program ?

(A) For
(B) While
(C) Do-While
(D) All of these

Correct Answer: (C) Do-While

Solution:

Step 1: Understanding the Concept:
A menu-driven program displays a list of options to the user and performs actions based on the user's choice. The program typically continues to display the menu until the user selects

an option to exit. This requires a loop structure.

Step 2: Detailed Explanation:
Let's evaluate the suitability of each loop for this task:

- **(A) For loop:** This loop is best used when the number of iterations is known in advance. It is not ideal for a menu that should run an indefinite number of times based on user input.

- **(B) While loop:** A 'while' loop checks its condition *before* executing the loop body. It can be used, but it often requires initializing the choice variable before the loop begins to ensure the loop is entered the first time.

- **(C) Do-While loop:** A 'do-while' loop executes its loop body *at least once* and then checks the condition at the end. This is the most natural fit for a menu-driven program because you always want to display the menu at least once. The structure is: "do {display menu and get choice}, while (choice is not 'exit')".

- **(D) All of these:** While it's technically possible to implement a menu with any loop, the 'do-while' loop's structure is the most direct and elegant solution for the problem, making it the "most suitable".

Step 3: Final Answer:
The 'do-while' loop is the most suitable control structure for a typical menu-driven program because it guarantees the menu is displayed at least once.

Quick Tip

Use a 'do-while' loop whenever you need to execute a block of code at least once. It's perfect for user input scenarios where you need to get input first and then validate it or check if the user wants to continue.

---

10. Which of the examples below expresses the commutative law of multiplication ?

(A) A + B = B + A
(B) AB = B + A
(C) AB = BA
(D) AB = A * B

Correct Answer: (C) AB = BA

Solution:

Step 1: Understanding the Concept:
The commutative law is a fundamental property in mathematics and algebra which states that the order of the operands does not affect the result of the operation. For multiplication, this law is expressed as $a \times b = b \times a$. In Boolean algebra, multiplication corresponds to the logical AND operation.

Step 2: Detailed Explanation:
Let's analyze the given options in the context of Boolean algebra:

- **(A) A + B = B + A:** This expresses the **commutative law of addition** (the logical OR operation).
- **(B) AB = B + A:** This equation is not a standard law of algebra.
- **(C) AB = BA:** This correctly expresses the **commutative law of multiplication** (the logical AND operation). It states that the order in which you AND two variables does not change the outcome.
- **(D) AB = A * B:** This equation simply shows two different notations for the same operation. 'AB' (juxtaposition) and 'A * B' both represent the multiplication (AND) of A and B. It does not express a law.

Step 3: Final Answer:
The expression that represents the commutative law of multiplication is 'AB = BA'.

Quick Tip

Remember the three main laws of Boolean algebra:

- **Commutative:** Order doesn't matter (A+B = B+A; AB = BA).
- **Associative:** Grouping doesn't matter (A+(B+C) = (A+B)+C; A(BC) = (AB)C).
- **Distributive:** A(B+C) = AB + AC.

---

11. Simplify: y = AB' + (A' + B)C

(A) AB' + C
(B) AB + AC
(C) A'B + AC'
(D) AB + A

Correct Answer: (A) AB' + C

Solution:

Step 1: Understanding the Concept:
This problem requires simplifying a Boolean expression using the laws of Boolean algebra, such as the distributive law and complementation laws.

Step 2: Key Formula or Approach:
The key laws to be used are:
1. **Distributive Law:** $X(Y + Z) = XY + XZ$ and $X + YZ = (X + Y)(X + Z)$.
2. **Complementation Law:** $X + X' = 1$.
3. **Identity Law:** $X \cdot 1 = X$.
4. **Adjacency/Absorption Law:** $X + X'Y = X + Y$.

Step 3: Detailed Explanation:
We are given the expression:

$$y = AB' + (A' + B)C$$

First, apply the distributive law to the second term:

$$y = AB' + A'C + BC$$

Now, we can use the consensus theorem ($XY + X'Z + YZ = XY + X'Z$), but a more direct approach is to use other laws. Let's use the property $X + YZ = (X + Y)(X + Z)$ to manipulate the expression. A simpler method is as follows:
From the expression $y = AB' + A'C + BC$, we can introduce the term $A + A' = 1$:

$$y = AB' + A'C + BC(A + A')$$
$$y = AB' + A'C + ABC + A'BC$$

Now, group the terms:

$$y = (AB' + ABC) + (A'C + A'BC)$$

Factor out common terms:

$$y = A(B' + BC) + A'C(1 + B)$$

Using the law $X' + XY = X' + Y$, we have $B' + BC = B' + C$. Also, $1 + B = 1$.

$$y = A(B' + C) + A'C(1)$$
$$y = AB' + AC + A'C$$

Factor out C from the last two terms:

$$y = AB' + C(A + A')$$

Since $A + A' = 1$:

$$y = AB' + C(1)$$

$$y = AB' + C$$

Step 4: Final Answer:
The simplified expression is $AB' + C$.

Quick Tip

When simplifying Boolean expressions, look for opportunities to apply the distributive law first. After expanding, try to group terms to factor out variables or use theorems like consensus ($XY + X'Z + YZ = XY + X'Z$) or absorption ($X + XY = X$).

---

12. If a * b = a + b + ab, then 3 * 4 - 2 * 3 is equal to

(A) 6
(B) 8
(C) 10
(D) 12

Correct Answer: (B) 8

Solution:

Step 1: Understanding the Concept:
The problem defines a new binary operator '*' and asks to evaluate an expression using this definition. We need to substitute the values into the given formula and perform the calculations in the correct order.

Step 2: Key Formula or Approach:
The defined operation is: $a * b = a + b + ab$.
We need to calculate $(3 * 4) - (2 * 3)$.

Step 3: Detailed Explanation:
First, calculate the value of $3 * 4$.
Here, $a = 3$ and $b = 4$.

$$3 * 4 = 3 + 4 + (3 \times 4) = 7 + 12 = 19$$

Next, calculate the value of $2 * 3$.
Here, $a = 2$ and $b = 3$.

$$2 * 3 = 2 + 3 + (2 \times 3) = 5 + 6 = 11$$

Finally, substitute these results back into the original expression:

$$(3 * 4) - (2 * 3) = 19 - 11 = 8$$

Step 4: Final Answer:
The value of the expression is 8.

Quick Tip

For questions with custom-defined operators, carefully substitute the numbers into the given formula. Pay close attention to the order of operations (PEMDAS/BODMAS) for the standard arithmetic part of the formula.

---

13. The topology of Ethernet system is

(A) Ring
(B) Star
(C) Mixed
(D) Bus

Correct Answer: (B) Star

Solution:

Step 1: Understanding the Concept:
Network topology refers to the physical arrangement of devices in a computer network. Ethernet is a widely used technology for local area networks (LANs).

Step 2: Detailed Explanation:

- **Early Ethernet (Legacy):** The original Ethernet standards (like 10BASE5 and 10BASE2) used a **Bus topology**, where all devices shared a single coaxial cable.
- **Modern Ethernet:** Since the introduction of twisted-pair cabling (like 10BASE-T, 100BASE-TX, Gigabit Ethernet), Ethernet has predominantly used a **Star topology**. In this setup, each device connects to a central hub or, more commonly, a switch.
- Larger networks often use an extended star or tree topology, which is a hybrid model but is fundamentally built from interconnected star topologies.

Given that star topology is the basis for virtually all modern Ethernet installations, it is considered the standard topology for current Ethernet systems.

Step 3: Final Answer:
While early Ethernet used a bus topology, the modern and most common topology for Ethernet

systems is the Star topology.

Quick Tip

When asked about a technology that has evolved, the question usually refers to its modern, most common implementation unless specified otherwise. For Ethernet, think of computers connected to a central switch, which is the classic star layout.

---

14. A set of rules is

(A) Resource locator
(B) Hypertext
(C) Protocol
(D) Domain

Correct Answer: (C) Protocol

Solution:

Step 1: Understanding the Concept:
This question asks for the specific term used in computing and networking to describe a set of rules.

Step 2: Detailed Explanation:

- **(A) Resource locator:** A URL (Uniform Resource Locator) is an address used to locate a resource on the internet. It's an address, not a set of rules.

- **(B) Hypertext:** This is text that contains links to other texts. HTML (HyperText Markup Language) is used to create it. It's a type of content, not a set of rules.

- **(C) Protocol:** A protocol is a formal set of rules and conventions that governs how data is formatted, transmitted, and received between computing devices. Examples include HTTP, TCP/IP, and FTP. This is the correct definition.

- **(D) Domain:** A domain (e.g., google.com) is a human-readable name that corresponds to an IP address, used for identifying and accessing websites. It's a naming system, not a set of rules.

Step 3: Final Answer:
A set of rules that governs communication between devices is called a protocol.

Quick Tip

Think of a protocol as a language that two computers agree to speak. Just like humans need shared rules of grammar and vocabulary to communicate, computers need protocols (like TCP/IP) to exchange information reliably.

---

15. In which medium is transmission speed the fastest ?

(A) Coaxial cable
(B) Microwave transmission
(C) Radio waves
(D) Optical fibre

Correct Answer: (D) Optical fibre

Solution:

Step 1: Understanding the Concept:
Transmission speed (or data rate) refers to the amount of data that can be transmitted through a communication medium in a given amount of time. This is largely determined by the medium's bandwidth and the physical properties of the signal being transmitted.

Step 2: Detailed Explanation:

- **(A) Coaxial cable:** Transmits electrical signals over a copper conductor. It offers higher bandwidth than twisted-pair cables but is limited by electrical resistance and interference.

- **(B) Microwave transmission & (C) Radio waves:** These are forms of wireless communication that transmit electromagnetic waves through the air. Their speed can be high but is susceptible to interference, distance, and atmospheric conditions. The speed of the wave is the speed of light, but the data rate (bandwidth) is limited.

- **(D) Optical fibre:** Transmits data as pulses of light through thin strands of glass. Since light has a very high frequency, optical fiber offers extremely high bandwidth. This allows for data transmission speeds that are significantly faster than any of the other options, reaching speeds of terabits per second (Tbps) over long distances.

Step 3: Final Answer:
Optical fibre provides the fastest transmission speeds due to its high bandwidth and the use of light as the transmission signal.

Quick Tip

Remember that light is the fastest thing in the universe. Media that use light to transmit data, like optical fiber, will generally offer the highest potential speeds and bandwidth compared to media that use electrical signals or radio waves.

---

16. A data structure in which elements can be inserted or deleted at / from both ends but not in the middle is

(A) Priority queue
(B) Dequeue
(C) Circular queue
(D) Queue

Correct Answer: (B) Dequeue

Solution:

Step 1: Understanding the Concept:
The question describes a specific type of linear data structure based on its insertion and deletion rules.

Step 2: Detailed Explanation:

- **(A) Priority queue:** An abstract data type where each element has a "priority". Elements with higher priority are served before elements with lower priority, regardless of their insertion order.
- **(B) Dequeue:** Short for "double-ended queue". It is a generalization of a queue in which elements can be added (enqueued) and removed (dequeued) from either the front or the rear end. This perfectly matches the description.
- **(C) Circular queue:** An implementation of a standard queue that uses a fixed-size array as if it were connected end-to-end. It still follows the FIFO (First-In, First-Out) principle, with insertion at the rear and deletion from the front.
- **(D) Queue:** A standard queue follows the FIFO principle. Elements are inserted at one end (rear) and deleted from the other end (front).

Step 3: Final Answer:
The data structure that allows insertion and deletion at both ends is a dequeue.

Quick Tip

Break down the word "Dequeue" into "D-E-Queue," which stands for Double-Ended Queue. This makes it easy to remember that it's a queue where you can perform operations at both ends.

---

17. In simple chaining, what data structure is appropriate ?

(A) Double linked list
(B) Circular linked list
(C) Single linked list
(D) Binary tree

Correct Answer: (C) Single linked list

Solution:

Step 1: Understanding the Concept:
Chaining is a common collision resolution technique used in hash tables. When two or more keys hash to the same index in the hash table array, these keys are stored in a separate data structure at that index.

Step 2: Detailed Explanation:

- In the "simple chaining" method, each slot in the hash table array points to a list of all the entries that hash to that slot.

- The most straightforward and common data structure for implementing these lists is a **singly linked list**.

- A singly linked list is sufficient because we typically just need to traverse the list to find, delete, or insert an element. Insertion is often done at the head of the list, which is an O(1) operation.

- While a doubly linked list could be used, the extra pointer (to the previous node) is usually unnecessary for this application, adding needless memory overhead. A binary tree (specifically a self-balancing one) could also be used to improve worst-case search time, but this is a more complex technique than "simple" chaining.

Step 3: Final Answer:
A singly linked list is the most appropriate and common data structure for simple chaining in hash tables.

Quick Tip

For hash table collision resolution, remember the two main strategies: Chaining (using a data structure, typically a linked list, to hold colliding items) and Open Addressing (probing for the next empty slot in the table itself). Simple chaining almost always implies using a singly linked list.

---

18. Which data type is used in SQL to store true or false values ?

(A) int
(B) Boolean
(C) Varchar
(D) Float

Correct Answer: (B) Boolean

Solution:

Step 1: Understanding the Concept:
The question asks for the standard SQL data type designed specifically for storing logical values, which are typically *true*, *false*, or sometimes *unknown*.

Step 2: Detailed Explanation:

- **(A) int:** An integer type. While many database systems allow using an integer (like '1' for true and '0' for false) to represent boolean values, it is not the dedicated data type for this purpose.
- **(B) Boolean:** The standard SQL data type for storing boolean values is 'BOOLEAN'. It can hold the values 'TRUE', 'FALSE', or 'NULL'.
- **(C) Varchar:** This type is used for storing variable-length character strings, like 'true' or 'false', which is inefficient and not standard practice.
- **(D) Float:** This type is for storing approximate-value numeric data (floating-point numbers).

Note: Some specific database systems like MySQL historically used 'TINYINT(1)' as a substitute for a true boolean type, but 'BOOLEAN' is the correct standard type.

Step 3: Final Answer:
The 'Boolean' data type is used in SQL to store true or false values.

Quick Tip

In SQL, always choose the data type that most accurately represents the data you are storing. For true/false or yes/no flags, the ‘BOOLEAN‘ type is the most appropriate and semantically correct choice.

---

19. In SQL, which data type is best suited for storing large texts such as articles or comments ?

(A) Text
(B) Varchar
(C) Char
(D) Blob

Correct Answer: (A) Text

Solution:

Step 1: Understanding the Concept:
SQL provides various data types for storing strings. The choice depends on the length and nature of the text. The question asks for the best type for large, variable-length text.

Step 2: Detailed Explanation:

- **(A) TEXT:** This data type is specifically designed for storing large blocks of text, like articles, comments, or book chapters. It can hold a very large number of characters (e.g., up to 65,535 bytes for ‘TEXT‘ in MySQL, with longer variants like ‘MEDIUMTEXT‘ and ‘LONGTEXT‘ available).

- **(B) VARCHAR(n):** This type is for variable-length strings, but you must specify a maximum length ‘n‘. It is suitable for shorter texts like names or titles, but is generally not the best choice for very large texts as it may have a lower maximum length compared to ‘TEXT‘ types.

- **(C) CHAR(n):** This type is for fixed-length strings. It wastes space if the text stored is shorter than the specified length ‘n‘, making it very unsuitable for variable-length articles or comments.

- **(D) BLOB (Binary Large Object):** This type is for storing large binary data, such as images, audio, or video files. It is not intended for storing plain text.

Step 3: Final Answer:
The ‘TEXT‘ data type is the most suitable for storing large text content like articles.

Quick Tip

Use this simple guide for SQL string types:

- ‘CHAR‘: For short, fixed-length strings (e.g., state abbreviations ’NY’, ’CA’).
- ‘VARCHAR‘: For short, variable-length strings (e.g., names, email addresses).
- ‘TEXT‘: For long, variable-length text (e.g., comments, articles).
- ‘BLOB‘: For binary data (e.g., images, files).

---

20. Which SQL command is used to create a new table in a database ?

(A) Create
(B) Alter
(C) Drop
(D) Truncate

Correct Answer: (A) Create

Solution:

Step 1: Understanding the Concept:
SQL commands are categorized into groups like DDL (Data Definition Language), DML (Data Manipulation Language), etc. Creating a table is a task that defines the database structure, so it falls under DDL.

Step 2: Detailed Explanation:

- **(A) CREATE:** The ‘CREATE‘ command is a DDL command used to create new database objects, including tables, views, and indexes. The specific syntax is ‘CREATE TABLE table_name (...)‘. This is the correct answer.

- **(B) ALTER:** The ‘ALTER TABLE‘ command is used to modify the structure of an existing table (e.g., add, delete, or modify columns).

- **(C) DROP:** The ‘DROP TABLE‘ command is used to permanently delete an existing table and all its data.

- **(D) TRUNCATE:** The ‘TRUNCATE TABLE‘ command is used to quickly remove all rows from a table, but it does not delete the table structure itself.

Step 3: Final Answer:
The ‘CREATE‘ command is used to create a new table in a database.

Quick Tip

Remember the main DDL commands with the acronym CAD: Create, Alter, Drop. These commands define and manage the *structure* of your database objects.

---

21. The size of int datatype in C++ is

(A) 1 byte
(B) 2 bytes
(C) 4 bytes
(D) Depends on compiler

Correct Answer: (D) Depends on compiler

Solution:

Step 1: Understanding the Concept:
The C++ language standard defines minimum ranges for its fundamental data types but does not mandate their exact size in bytes. The size is left up to the implementation (the compiler and the target machine architecture).

Step 2: Detailed Explanation:

- The C++ standard guarantees that ‘sizeof(char) ¡= sizeof(short) ¡= sizeof(int) ¡= sizeof(long) ¡= sizeof(long long)‘.

- It also guarantees that an ‘int‘ must be able to hold values at least in the range [-32767, +32767], which implies a minimum size of 16 bits (2 bytes).

- On older 16-bit systems (like MS-DOS), an ‘int‘ was typically 2 bytes.

- On most modern 32-bit and 64-bit systems, an ‘int‘ is typically 4 bytes.

- Because the size is not fixed and varies between different platforms and compilers, the most accurate answer is that it ”depends on the compiler” and the underlying architecture.

Step 3: Final Answer:
The size of the ‘int‘ datatype in C++ is platform-dependent and is determined by the compiler.

Quick Tip

Never assume a fixed size for fundamental types like ‘int‘ in C++ if you are writing portable code. If you need a variable of a specific size (e.g., exactly 32 bits), use the fixed-width integer types from the ‘¡cstdint¿‘ header, such as ‘int32$_t‘or‘uint32_t$‘.

---

22. Which of the following is not the type of queue?

(A) Priority queue
(B) Single ended queue
(C) Circular queue
(D) Ordinary queue

Correct Answer: (B) Single ended queue

Solution:

Step 1: Understanding the Concept:
A queue is a linear data structure that follows the First-In, First-Out (FIFO) principle. There are several variations and implementations of this basic concept. The question asks to identify the term that is not a standard type of queue.

Step 2: Detailed Explanation:

- **(A) Priority Queue:** This is a well-known abstract data type. It is a type of queue where elements are dequeued based on their priority rather than their arrival time. This is a standard type.

- **(C) Circular Queue:** This is a standard and efficient way to implement a queue using a fixed-size array. It overcomes the limitations of a simple array-based queue by treating the array as circular. This is a standard implementation type.

- **(D) Ordinary Queue:** This is a common term used to refer to a basic, simple, FIFO queue, to distinguish it from more complex variations like a priority queue. It is a valid description.

- **(B) Single ended queue:** This term is not standard terminology in data structures. A standard queue has two distinct ends: a ‘front‘ for deletion and a ‘rear‘ for insertion. A data structure with operations restricted to a single end is a 'stack'. The term "single-ended queue" is ambiguous and not a recognized type. While it might be used informally to contrast with a 'deque' (double-ended queue), it's the least standard and most likely incorrect term among the choices.

Note: While the checkmark in the source image appears to be on (D), "Single ended queue" is technically the non-standard term. The most logical answer based on computer science terminology is (B). We will proceed with the most technically correct answer.

Step 3: Final Answer:
"Single ended queue" is not a standard recognized type of queue data structure.

Quick Tip

Be familiar with the main queue variations: Simple/Ordinary Queue (FIFO), Circular Queue (implementation), Priority Queue (priority-based), and Deque (Double-Ended Queue). Any term outside of these is likely to be incorrect or non-standard.

---

23. In SQL which of the following is a data definition language command?

(A) Alter Table
(B) Revoke
(C) Grant
(D) Update

Correct Answer: (A) Alter Table

Solution:

Step 1: Understanding the Concept:
SQL commands are categorized based on their function. Data Definition Language (DDL) commands are used to define and manage the structure of database objects.

Step 2: Detailed Explanation:
Let's classify the given commands:

- **(A) Alter Table:** This command is used to modify the structure of an existing table (e.g., add or drop columns). Since it deals with the table's definition, it is a **DDL** command.

- **(B) Revoke:** This command is used to remove user access permissions. It is part of the Data Control Language (**DCL**).

- **(C) Grant:** This command is used to give users access permissions to database objects. It is also a **DCL** command.

- **(D) Update:** This command is used to modify existing data within a table. Since it manipulates the data itself, it is a Data Manipulation Language (**DML**) command.

Step 3: Final Answer:
The only DDL command listed is ‘Alter Table‘.

Quick Tip

Remember the main SQL command categories:

- **DDL (Definition):** CREATE, ALTER, DROP, TRUNCATE
- **DML (Manipulation):** SELECT, INSERT, UPDATE, DELETE
- **DCL (Control):** GRANT, REVOKE

---

24. Which is the data type of text ?

(A) var char
(B) var char ignore case
(C) short var char
(D) Both (A) and (B)

Correct Answer: (A) var char

Solution:

Step 1: Understanding the Concept:
The question asks to identify a valid data type used for storing text in databases like SQL.

Step 2: Detailed Explanation:

- **(A) var char:** This is the standard SQL keyword for the ‘VARCHAR‘ data type, which is used to store variable-length character strings. This is a correct and standard data type for text.

- **(B) var char ignore case:** This is not a standard SQL data type name. Case sensitivity is typically handled by ‘collations‘ (a set of rules for comparing characters) applied to a column, not by the data type name itself.

- **(C) short var char:** This is not a standard SQL data type.

Step 3: Final Answer:
Among the given options, 'var char' ('VARCHAR') is the standard SQL data type for text.

Quick Tip

Focus on standard SQL data types. Common text types are 'CHAR', 'VARCHAR', and 'TEXT'. Terms like "ignore case" or "short" are usually not part of the data type's name itself.

---

25. Internet is a

(A) Collection of WAN
(B) Network of Networks
(C) Collection of Networks
(D) Collection of identical LAN & MAN

Correct Answer: (B) Network of Networks

Solution:

Step 1: Understanding the Concept:
This question asks for the best definition of the Internet from the given choices.

Step 2: Detailed Explanation:

- The Internet is a massive, global network that connects millions of smaller computer networks. These smaller networks can be of any type—Local Area Networks (LANs), Wide Area Networks (WANs), Metropolitan Area Networks (MANs)—and they are operated by various entities (governments, companies, universities, etc.).

- **(A) Collection of WAN:** This is too specific. The Internet includes many other types of networks besides WANs.

- **(B) Network of Networks:** This is the classic and most accurate definition. It perfectly describes the architecture of the Internet as a system that interconnects countless disparate networks.

- **(C) Collection of Networks:** This is also true, but "Network of Networks" is a more precise term because it emphasizes the *interconnection* that allows them to communicate and function as a single global system.

- **(D) Collection of identical LAN & MAN:** This is incorrect. The networks connected to the Internet are highly diverse and not identical.

Step 3: Final Answer:
The best and most standard definition of the Internet is a "Network of Networks."

Quick Tip

The word "Internet" comes from "inter-networking." This etymology directly points to its definition as a network that connects other networks together.

---

26. On which of the following OSI layer is checksum not operable ?

(A) Session layer
(B) Transport layer
(C) Network layer
(D) Data link layer

Correct Answer: (A) Session layer

Solution:

Step 1: Understanding the Concept:
The OSI (Open Systems Interconnection) model describes the functions of a networking system in seven layers. A checksum is a form of error detection. We need to identify which layer does not typically perform error checking using checksums.

Step 2: Detailed Explanation:

- **(D) Data Link Layer (Layer 2):** This layer is responsible for error-free transfer of data frames between nodes on the same physical link. It uses a Frame Check Sequence (FCS), often a Cyclic Redundancy Check (CRC), which is a type of checksum, to detect corrupted frames.

- **(C) Network Layer (Layer 3):** The header of an IPv4 packet contains a checksum field that is used to check for errors in the header itself.

- **(B) Transport Layer (Layer 4):** Protocols like TCP and UDP use a checksum to verify the integrity of both the header and the data payload of their segments. This provides end-to-end error checking.

- **(A) Session Layer (Layer 5):** This layer is responsible for establishing, managing, and terminating sessions between applications. It manages dialogue control and synchronization. It does not deal with data integrity at the level of checksums; that function is left to the lower layers, primarily the Transport and Data Link layers.

Step 3: Final Answer:
The Session layer is not typically responsible for error detection using checksums.

Quick Tip

Remember the primary roles of the OSI layers. Error detection and correction are key functions of the Data Link (Layer 2) and Transport (Layer 4) layers. The upper layers (Session, Presentation, Application) generally rely on these lower layers to provide a reliable data stream.

---

27. In SQL which command is used to change data in a table ?

(A) Update
(B) Insert
(C) Browse
(D) None of these

Correct Answer: (A) Update

Solution:

Step 1: Understanding the Concept:
This question asks for the specific SQL command used to modify existing records (rows) in a database table. This is a fundamental data manipulation task.

Step 2: Detailed Explanation:

- **(A) UPDATE:** The 'UPDATE' command is the standard SQL command used to modify the values in one or more existing rows of a table. You typically use a 'WHERE' clause to specify which rows to change. This is the correct answer.

- **(B) INSERT:** The 'INSERT INTO' command is used to add new rows of data to a table. It does not change existing data.

- **(C) Browse:** "Browse" is not a standard SQL command. It might be a feature in a graphical database client to view data, but it is not part of the SQL language itself.

- **(D) None of these:** This is incorrect as ‘UPDATE‘ is the correct command.

Step 3: Final Answer:
The ‘UPDATE‘ command is used to change existing data in a table.

Quick Tip

To remember the basic DML commands, think about what you can do to data rows:

- ‘INSERT‘ new rows.
- ‘UPDATE‘ existing rows.
- ‘DELETE‘ existing rows.
- ‘SELECT‘ (read) existing rows.

---

28. Which command is used to create a new row in SQL ?

(A) Insert Into
(B) Alter Table
(C) Update
(D) All of these

Correct Answer: (A) Insert Into

Solution:

Step 1: Understanding the Concept:
The question asks for the SQL command used to add new records, or rows, to a database table.

Step 2: Detailed Explanation:

- **(A) INSERT INTO:** This is the correct and complete SQL command for adding one or more new rows of data into a specified table.

- **(B) ALTER TABLE:** This command is used to change the structure (definition) of a table, such as adding or removing columns, not rows of data.

- **(C) UPDATE:** This command is used to modify data in rows that already exist in the table.

- **(D) All of these:** This is incorrect as only ‘INSERT INTO‘ is used for creating new rows.

Step 3: Final Answer:
The ‘INSERT INTO‘ command is used to create a new row in an SQL table.

Quick Tip

Differentiate between changing the data and changing the structure. ‘INSERT‘, ‘UPDATE‘, ‘DELETE‘ (DML) commands change the *data* (the rows). ‘ALTER‘ (DDL) command changes the *structure* (the columns and constraints).

---

29. Using which of the following keywords can an exception be generated ?

(A) Threw
(B) Throws
(C) Throw
(D) Catch

Correct Answer: (C) Throw

Solution:

Step 1: Understanding the Concept:
Exception handling is a mechanism in programming languages to deal with runtime errors. The process involves ”throwing” an exception when an error occurs and ”catching” it to handle the error. The question asks for the keyword that actively generates an exception.

Step 2: Detailed Explanation:

- **(C) throw:** In languages like C++, Java, and C#, the ‘throw‘ keyword is used to signal that an exceptional condition has occurred. It creates an exception object and hands it over to the runtime system to find a suitable handler. Example: ‘throw new Exception(”Error occurred”);‘.

- **(A) Threw:** This is the past tense of the verb ”throw” and is not a keyword in programming.

- **(B) throws:** In Java, the ‘throws‘ keyword is used in a method’s signature to declare the types of exceptions that the method might throw. It does not generate the exception itself but informs the callers of the method about potential exceptions they need to handle.

- **(D) catch:** The ‘catch‘ keyword is used to define a block of code (a handler) that is executed when a specific type of exception is thrown in the associated ‘try‘ block. It handles an exception, it does not generate one.

Step 3: Final Answer:
The ‘throw‘ keyword is used to generate an exception.

Quick Tip

Remember the flow: You ‘try‘ some code. If an error occurs, you ‘throw‘ an exception. A ‘catch‘ block then handles that exception. The ‘throws‘ keyword (in Java) is a warning sign on a method’s door, saying ”I might throw these exceptions.”

---

30. What is the ASCII value of ’0’ character ?

(A) 32
(B) 24
(C) 48
(D) 0

Correct Answer: (C) 48

Solution:

Step 1: Understanding the Concept:
ASCII (American Standard Code for Information Interchange) is a character encoding standard that assigns a unique numeric value to each character. The question asks for the decimal value corresponding to the character ’0’.

Step 2: Detailed Explanation:
In the standard ASCII table:

- The decimal value for the character ’0’ is **48**.

- The characters ’1’ through ’9’ follow sequentially, with ’1’ being 49, ’2’ being 50, and so on, up to ’9’ which is 57.

- For reference, the decimal value for a space character (' ') is 32.
- The decimal value for the uppercase letter 'A' is 65.
- The decimal value for the lowercase letter 'a' is 97.

Step 3: Final Answer:
The ASCII decimal value of the character '0' is 48.

Quick Tip

It's useful to memorize a few key ASCII values for competitive exams: '0' is 48, 'A' is 65, and 'a' is 97. You can derive the value of other letters and digits from these benchmarks (e.g., 'C' is 'A'+2 = 65+2 = 67).

---

31. Which of the following data types is supported in C++ but not in C ?

(A) int
(B) bool
(C) double
(D) float

Correct Answer: (B) bool

Solution:

Step 1: Understanding the Concept:
C++ was developed as an extension of the C language, adding new features and data types. This question asks to identify a data type that is native to C++ but was not part of the original C language.

Step 2: Detailed Explanation:

- **(A) int, (C) double, (D) float:** These are fundamental numeric data types that are present in both C and C++.
- **(B) bool:** The ‘bool‘ data type, which can hold the values ‘true‘ and ‘false‘, is a built-in type in C++. In traditional C (ANSI C/C89), there was no boolean type. Programmers simulated it using integers (where ‘0‘ meant false and any non-zero value meant true). While the later C99 standard introduced a ‘_Bool‘ type and the ‘¡stdbool.h¿‘ header to define ‘bool‘, it is fundamentally a native keyword and concept in C++ that was absent

from C for a long time.

Step 3: Final Answer:
The ‘bool‘ data type is supported natively in C++ but not in the original C standard (C89/90).

Quick Tip

The key additions of C++ over C include Object-Oriented Programming features (classes, inheritance), templates, exceptions, and new built-in types like ‘bool‘ and ‘string‘. ‘bool‘ is a classic example of a type that C++ has but C originally lacked.

---

32. What is the full form of Telnet ?

(A) Telephone Network
(B) Teletype Network
(C) Teleconnect Network
(D) Television Network

Correct Answer: (B) Teletype Network

Solution:

Step 1: Understanding the Concept:
Telnet is one of the earliest network protocols used on the Internet. The question asks for the expansion of its name.

Step 2: Detailed Explanation:

- **Telnet** stands for **Teletype Network**.
- It was designed to provide a bidirectional, text-based communication channel between two machines over a network. It functions as a virtual terminal, allowing a user to log into a remote computer and execute commands as if they were sitting at that machine.
- The name reflects its purpose of simulating a "teletype" or "teleprinter" terminal over a network connection.
- The other options are incorrect expansions.

Step 3: Final Answer:
The full form of Telnet is Teletype Network.

Quick Tip

Telnet is an important historical protocol, but it is now considered insecure because it transmits all data, including passwords, in plain text. For secure remote command-line access, protocols like SSH (Secure Shell) are used instead.

---

33. Which of the following is not a part of an e-mail address ?

(A) User name
(B) Domain name
(C) @
(D) Protocol

Correct Answer: (D) Protocol

Solution:

Step 1: Understanding the Concept:
An email address has a standard format that uniquely identifies a user's mailbox on a network. The question asks to identify which of the given options is not a component of this standard format.

Step 2: Detailed Explanation:
The standard structure of an email address is 'username@domain.com'.

- **(A) User name:** This is the part before the '@' symbol. It identifies the specific user or mailbox. For example, in 'example@mail.com', "example" is the user name.
- **(B) Domain name:** This is the part after the '@' symbol. It specifies the mail server that hosts the user's mailbox. In the example, "mail.com" is the domain name.
- **(C) @:** The "at" symbol is the separator between the user name and the domain name. It is a required part of every email address.
- **(D) Protocol:** A protocol, such as SMTP (Simple Mail Transfer Protocol) or POP3 (Post Office Protocol 3), is a set of rules used to send and receive emails. It is part of the underlying email system's operation but is not a component of the email address itself.

Step 3: Final Answer:
Therefore, a protocol is not a part of an email address.

Quick Tip

Think of an email address like a postal address. The ‘username‘ is the recipient’s name, the ‘domain name‘ is the city and street, and the ‘@‘ symbol separates them. The ‘protocol‘ is like the postal service itself—it’s the system that delivers the mail, not part of the address written on the envelope.

---

34. In which year was world wide web started ?

(A) 1968
(B) 1969
(C) 1989
(D) 1985

Correct Answer: (C) 1989

Solution:

Step 1: Understanding the Concept:
The question asks for the year when the World Wide Web (WWW) was created. It is important to distinguish the World Wide Web from the Internet itself.

Step 2: Detailed Explanation:

- **1969:** This year marks the creation of ARPANET, the precursor to the modern Internet.
- **1989:** In March 1989, Sir Tim Berners-Lee, a British scientist at CERN, wrote a proposal for what would become the World Wide Web. He developed the key technologies for the Web, including HTML, URL, and HTTP, over the next couple of years. The year of its proposal and invention is recognized as 1989.

The other years listed are not significant milestones in the creation of the World Wide Web.

Step 3: Final Answer:
The World Wide Web was started in 1989 with the proposal by Tim Berners-Lee.

Quick Tip

Don't confuse the Internet with the World Wide Web. The Internet is the global network of computers, which started with ARPANET in 1969. The Web is a system of interlinked hypertext documents accessed via the Internet, which was invented in 1989.

---

35. How many bits are in IPv6 address ?

(A) 32 bits
(B) 128 bits
(C) 32 bytes
(D) 128 bytes

Correct Answer: (B) 128 bits

Solution:

Step 1: Understanding the Concept:
IP (Internet Protocol) addresses are numerical labels assigned to each device connected to a computer network. There are two main versions in use: IPv4 and IPv6. The question asks for the size of an IPv6 address.

Step 2: Detailed Explanation:

- **IPv4 (Internet Protocol version 4):** Uses a **32-bit** address space. This allows for 2^{32} (about 4.3 billion) unique addresses.

- **IPv6 (Internet Protocol version 6):** Was developed to address the exhaustion of IPv4 addresses. It uses a **128-bit** address space, which allows for 2^{128} (a vastly larger number) of unique addresses.

- Let's check the options in bytes as well (1 byte = 8 bits):
  - (C) 32 bytes = 32 * 8 = 256 bits.
  - (D) 128 bytes = 128 * 8 = 1024 bits.

Step 3: Final Answer:
An IPv6 address is 128 bits long.

Quick Tip

To easily remember the address sizes: IPv4 is 32 bits. IPv6 is four times larger in bits, so $4 \times 32 = 128$ bits.

36. How many layers are there in the ISO OSI reference model ?

(A) 7
(B) 5
(C) 4
(D) 6

Correct Answer: (A) 7

Solution:

Step 1: Understanding the Concept:
The ISO OSI (International Organization for Standardization - Open Systems Interconnection) model is a conceptual framework that standardizes the functions of a telecommunication or computing system in terms of seven abstraction layers.

Step 2: Detailed Explanation:
The seven layers of the OSI model, from top to bottom, are:
7. **Application Layer**
6. **Presentation Layer**
5. **Session Layer**
4. **Transport Layer**
3. **Network Layer**
2. **Data Link Layer**
1. **Physical Layer**
Therefore, there are a total of 7 layers in the model.

Step 3: Final Answer:
The ISO OSI reference model has 7 layers.

Quick Tip

Use a mnemonic to remember the 7 layers. A popular one is ”All People Seem To Need Data Processing” (for Application, Presentation, Session, Transport, Network, Data Link, Physical) starting from Layer 7 downwards.

37. Which of the following is not a network edge device ?

(A) Switch
(B) PC
(C) Smartphone

(D) Server

Correct Answer: (A) Switch

Solution:

Step 1: Understanding the Concept:
A computer network can be conceptually divided into the ”network edge” and the ”network core”.

- **Network Edge Devices (End Systems):** These are the devices that users interact with to access network services. They are the sources and destinations of network traffic. Examples include PCs, smartphones, servers, etc.

- **Network Core Devices (Intermediary Devices):** These devices connect the edge devices and facilitate the routing and forwarding of data across the network. Examples include routers and switches.

Step 2: Detailed Explanation:

- **(B) PC, (C) Smartphone, (D) Server:** These are all end systems or hosts. They sit at the edge of the network, generating and receiving data.

- **(A) Switch:** A switch is a network core device. Its primary function is to forward data packets (frames) between devices on the same local network. It operates within the network core to connect the edge devices.

Step 3: Final Answer:
A switch is a network core device, not a network edge device.

Quick Tip

Think of it this way: Edge devices are the ”what” (what sends and receives data), while core devices are the ”how” (how the data gets from one edge to another). A PC sends an email; a switch helps forward it within the local network.

---

38. What is the term for the data communication system within a building and campus ?

(A) MAN
(B) LAN

(C) PAN
(D) WAN

Correct Answer: (B) LAN

Solution:

Step 1: Understanding the Concept:
Computer networks are classified by their geographical scope. The question asks to identify the network type that covers a limited area like a building or a university campus.

Step 2: Detailed Explanation:

- **(C) PAN (Personal Area Network):** Covers a very small area, typically around a single person (e.g., connecting a phone to earbuds via Bluetooth).

- **(B) LAN (Local Area Network):** Covers a limited geographical area such as a home, office building, or a university campus. This matches the question's description.

- **(A) MAN (Metropolitan Area Network):** Covers a larger area than a LAN, such as a city or a large town.

- **(D) WAN (Wide Area Network):** Covers a very large geographical area, such as a country or even the entire globe (e.g., the Internet).

Step 3: Final Answer:
A data communication system within a building and campus is called a LAN (Local Area Network).

> Quick Tip
>
> Remember the network types in order of increasing size: PAN <LAN <MAN <WAN.

---

39. What was the name of the first network ?

(A) ASAPNET
(B) ARPANET
(C) CNNET
(D) NSFNET

Correct Answer: (B) ARPANET

Solution:

Step 1: Understanding the Concept:
The question asks for the name of the first large-scale, packet-switching network, which is considered the precursor to the modern Internet.

Step 2: Detailed Explanation:

- **(B) ARPANET (Advanced Research Projects Agency Network):** This was the first operational packet-switching network and the predecessor to the global Internet. It was developed by the Advanced Research Projects Agency (ARPA) of the U.S. Department of Defense and became operational in 1969.

- **(D) NSFNET (National Science Foundation Network):** This was another important network that came later and formed a major backbone of the Internet in the 1980s and early 1990s, but it was not the first.

- The other options, ASAPNET and CNNET, are not recognized names of major early networks.

Step 3: Final Answer:
The name of the first major packet-switched network was ARPANET.

Quick Tip

Associate ARPANET with the "A" in "Advanced" and the start of the internet. It was the "A-team" that kicked off modern networking.

---

40. Stack is also called as

(A) First in first out
(B) Fist in last out
(C) Last in last out
(D) Last in first out

Correct Answer: (D) Last in first out

Solution:

Step 1: Understanding the Concept:
A stack is a fundamental linear data structure that operates on a specific principle for adding

and removing elements. The question asks for the name of this principle.

Step 2: Detailed Explanation:

- A stack follows the **Last-In, First-Out (LIFO)** principle. This means the last element added to the stack is the first one to be removed. Think of a stack of plates: you put a new plate on top and you take a plate from the top.

- **(A) First in first out (FIFO):** This describes a Queue data structure.

- **(B) Fist in last out:** This is a typo but likely intended to mean LIFO, however it is not the standard term.

- **(C) Last in last out (LILO):** This is another way of describing FIFO.

- **(D) Last in first out (LIFO):** This is the correct and standard term for the principle governing a stack.

Step 3: Final Answer:
A stack is also called a Last-In, First-Out (LIFO) data structure.

Quick Tip

To remember the difference, visualize:

- **Stack (LIFO):** A stack of books. You add to the top and remove from the top.
- **Queue (FIFO):** A line of people at a checkout. The first person in line is the first person to be served.

---

41. The printer can be accessed using the predefined file name

(A) Printer file
(B) Print
(C) Cout
(D) PRN: LPT1, etc.

Correct Answer: (D) PRN: LPT1, etc.

Solution:

Step 1: Understanding the Concept:
In older operating systems like MS-DOS, hardware devices were often treated as files and could be accessed using special, reserved file names. This question refers to these predefined names for the printer port.

Step 2: Detailed Explanation:

- **(D) PRN: LPT1, etc.:** In MS-DOS and early versions of Windows, ‘PRN‘ was the logical device name for the default printer. ‘LPT1‘ (Line Print Terminal 1) was the name for the first parallel port, to which printers were typically connected. These reserved names could be used in commands to send output directly to the printer (e.g., ‘COPY file.txt LPT1‘).

- **(A) Printer file:** This is a generic description, not a specific predefined name.

- **(B) Print:** This is a command used to print files, not the name of the device file itself.

- **(C) Cout:** This refers to the standard character output stream in C++, which typically directs output to the console, not the printer by default.

Step 3: Final Answer:
The printer could be accessed using predefined file names like ‘PRN‘ and ‘LPT1‘.

Quick Tip

This is a concept from legacy computing. Remember that in systems like DOS and Unix-like OS, the philosophy ”everything is a file” was common. This meant you could interact with devices like printers (‘LPT1‘) and the console (‘CON‘) as if they were files.

---

42. Which is the application of stack ?

(A) Function call
(B) Large number Arithmetic
(C) Evaluation of Arithmetic expressions
(D) All of these

Correct Answer: (D) All of these

Solution:

Step 1: Understanding the Concept:

The stack data structure, due to its LIFO (Last-In, First-Out) nature, is used in a wide variety of computing applications. The question asks to identify which of the given options are applications of a stack.

Step 2: Detailed Explanation:

- **(A) Function call:** When functions are called in most programming languages, their local variables and return addresses are pushed onto a system stack (the ”call stack”). When a function returns, its information is popped off the stack. This is a fundamental application.
- **(B) Large number Arithmetic:** Stacks can be used to perform arithmetic on numbers that are too large to fit into standard data types by storing digits of the numbers in stacks and processing them.
- **(C) Evaluation of Arithmetic expressions:** Stacks are crucial for converting infix expressions (e.g., ‘A + B * C‘) to postfix (e.g., ‘A B C * +‘) and for evaluating postfix expressions, as this process naturally follows the LIFO principle for handling operators and operands.
- **(D) All of these:** Since all the listed options are valid and common applications of the stack data structure, this is the correct choice.

Step 3: Final Answer:

Function calls, large number arithmetic, and the evaluation of arithmetic expressions are all applications of a stack.

Quick Tip

Other common stack applications include backtracking algorithms (like solving a maze), implementing ”undo” functionality in editors, and checking for balanced parentheses in code.

---

43. Which one is not related to DBMS ?

(A) Entity
(B) Attribute
(C) Class
(D) Row

Correct Answer: (C) Class

Solution:

Step 1: Understanding the Concept:
The question asks to identify a term that is not a fundamental concept in the context of a typical Database Management System (DBMS), particularly relational databases.

Step 2: Detailed Explanation:

- **(A) Entity:** In database design (specifically the Entity-Relationship model), an entity is an object of interest, such as a person, place, or thing. Entities become tables in a relational database.

- **(B) Attribute:** An attribute is a property or characteristic of an entity. Attributes become columns in a table.

- **(D) Row:** A row (also known as a tuple or record) in a table represents a single instance of an entity.

- **(C) Class:** A class is a fundamental concept in Object-Oriented Programming (OOP). It is a blueprint for creating objects. While Object-Oriented Databases (OODBMS) and Object-Relational Databases (ORDBMS) exist and use concepts like classes, ”Class” is not a core term in traditional relational DBMS theory, unlike Entity, Attribute, and Row.

Step 3: Final Answer:
”Class” is a term primarily related to Object-Oriented Programming, not traditional DBMS.

> Quick Tip
>
> Associate Entity, Attribute, and Row/Tuple with relational databases. Associate Class, Object, and Inheritance with Object-Oriented Programming.

---

44. Which one is the smallest computer network ?

(A) Personal Area Network
(B) LAN
(C) MAN
(D) WAN

Correct Answer: (A) Personal Area Network

Solution:

Step 1: Understanding the Concept:
Computer networks are categorized based on their geographical scale. The question asks for the network type with the smallest range.

Step 2: Detailed Explanation:
Let's compare the typical ranges:

- **(A) Personal Area Network (PAN):** Covers a very short distance, usually up to 10 meters, centered around an individual's workspace. Bluetooth and Zigbee are common PAN technologies.

- **(B) Local Area Network (LAN):** Spans a building or a campus, typically up to a few kilometers. Wi-Fi and Ethernet are LAN technologies.

- **(C) Metropolitan Area Network (MAN):** Spans a city, typically up to a few tens of kilometers.

- **(D) Wide Area Network (WAN):** Spans a large geographical area, such as a country or the globe. The Internet is the largest WAN.

Based on this comparison, the PAN is the smallest type of computer network.

Step 3: Final Answer:
The smallest computer network is the Personal Area Network (PAN).

Quick Tip

The names of the network types are a good clue to their size. "Personal" is smaller than "Local," which is smaller than "Metropolitan" (city-wide), which is smaller than "Wide."

---

45. Which of the following is the bitwise operator ?

(A) ++
(B) <<
(C) ##
(D) %%

Correct Answer: (B) <<

Solution:

Step 1: Understanding the Concept:
A bitwise operator is an operator used in programming to perform operations on individual bits of integer types. The question asks to identify a bitwise operator from the given choices.

Step 2: Detailed Explanation:

- **(A) ++:** This is the increment operator, which is an arithmetic operator.
- **(B) <<:** This is the **bitwise left shift** operator. It shifts the bits of its left operand to the left by the number of positions specified by its right operand. It is a bitwise operator.
- **(C) ##:** This is the token-pasting operator, used only within the C/C++ preprocessor for macro definitions. It is not a general-purpose operator.
- **(D) %%:** This is not a valid operator in C++. The single ‘

Step 3: Final Answer:
The bitwise operator among the choices is ‘¡¡‘ (left shift).

Quick Tip

Remember the main bitwise operators in C++: “ (AND), ‘—‘ (OR), ‘‘$(XOR),‘\ ‘(NOT),‘ << ‘(leftshift), and‘ >> ‘(rightshift)$.

---

46. Which one is non-procedural language ?

(A) C
(B) SQL
(C) Java
(D) C++

Correct Answer: (B) SQL

Solution:

Step 1: Understanding the Concept:
Programming languages can be broadly classified as procedural or non-procedural (declarative).

- **Procedural languages** require the programmer to specify an explicit sequence of steps (a procedure or algorithm) to achieve a result.

- **Non-procedural languages** (or declarative languages) require the programmer to specify *what* result they want, leaving the implementation of *how* to get it to the language's compiler or runtime.

Step 2: Detailed Explanation:

- **(A) C, (C) Java, (D) C++:** These are all primarily procedural (or imperative) languages. C is purely procedural, while Java and C++ are object-oriented but still require programmers to write step-by-step instructions within methods.

- **(B) SQL (Structured Query Language):** SQL is the classic example of a declarative language. A user writes a query like 'SELECT name FROM students WHERE grade = 'A';'. They are declaring the data they want, not providing a step-by-step algorithm for searching tables, creating indexes, or joining data. The database engine figures out the most efficient procedure to retrieve the requested data.

Step 3: Final Answer:
SQL is a non-procedural (declarative) language.

Quick Tip

Think of it as giving instructions. Procedural is like giving a recipe: "First, do this. Second, do that...". Non-procedural is like ordering from a menu: "I want a burger." You don't specify how the chef should cook it.

---

47. IP stands for

(A) Internet protocol
(B) www
(C) Intranet
(D) Internet post

Correct Answer: (A) Internet protocol

Solution:

Step 1: Understanding the Concept:
The question asks for the full form of the acronym IP, which is a fundamental component of the Internet's protocol suite.

Step 2: Detailed Explanation:

- **IP** stands for **Internet Protocol**. It is the principal communications protocol in the Internet protocol suite for relaying datagrams across network boundaries. Its primary function is addressing hosts and routing packets from a source host to a destination host across one or more IP networks.

- The other options are incorrect. ‘www‘ is the World Wide Web, and ‘Intranet‘ is a private network. ‘Internet post‘ is not a standard term.

Step 3: Final Answer:
IP stands for Internet Protocol.

Quick Tip

Remember the core internet suite is called TCP/IP, which stands for Transmission Control Protocol / Internet Protocol. This helps recall what both acronyms stand for.

---

48. How is an array initialized in C++ language ?

(A) int a [3] = {1, 2, 3};
(B) int a = {1, 2, 3};
(C) int a [ ] = new int [3];
(D) int a (3) = {1, 2, 3};

Correct Answer: (A) int a [3] = {1, 2, 3};

Solution:

Step 1: Understanding the Concept:
The question asks for the correct syntax to declare and initialize an array in C++ at the same time.

Step 2: Detailed Explanation:
Let’s analyze the syntax of each option:

- **(A) ‘int a [3] = {1, 2, 3};‘:** This is the correct and standard C-style syntax for initializing an array in C++. It declares an array named ‘a‘ of type ‘int‘ with a size of 3 and initializes it with the provided values.

- **(B) ‘int a = {1, 2, 3};‘:** This is incorrect. The ‘[]‘ are missing to denote that ‘a‘ is an array.

- **(C) ‘int a [ ] = new int [3];‘:** This is syntactically incorrect. The ‘new‘ operator allocates memory on the heap and returns a pointer. The correct syntax for dynamic allocation would be ‘int *a = new int[3];‘. Furthermore, this only allocates memory; it doesn’t initialize it with the values 1, 2, 3.

- **(D) ‘int a (3) = {1, 2, 3};‘:** This syntax is incorrect. Parentheses are not used for array declaration in this manner.

Step 3: Final Answer:
The correct way to initialize an array among the given options is ‘int a [3] = {1, 2, 3};‘.

Quick Tip

For array initialization in C++, remember the syntax: ‘type name[size] = value1, value2, ...;‘. You can also omit the size if you provide an initializer list, and the compiler will deduce the size, e.g., ‘int a[] = 1, 2, 3;‘.

---

49. Which of the following is a linear data structure ?

(A) Array
(B) AVL Tree
(C) Binary Tree
(D) Graph

Correct Answer: (A) Array

Solution:

Step 1: Understanding the Concept:
Data structures are classified as either linear or non-linear.

- **Linear Data Structures:** Elements are arranged sequentially or linearly, where each element is attached to its previous and next adjacent elements. Examples: Array, Linked List, Stack, Queue.

- **Non-Linear Data Structures:** Elements are arranged in a hierarchical or networked manner, where an element can be connected to multiple other elements. Examples: Tree, Graph.

Step 2: Detailed Explanation:

- **(A) Array:** An array stores elements in contiguous memory locations, forming a clear linear sequence. It is a linear data structure.

- **(B) AVL Tree (C) Binary Tree:** Trees are hierarchical data structures where each node can have multiple children. They are non-linear.

- **(D) Graph:** A graph consists of vertices and edges connecting them, representing a network. It is a non-linear data structure.

Step 3: Final Answer:
An array is a linear data structure.

Quick Tip

A simple test: If you can arrange all the elements in a single row or column without losing the structural relationship between them, it's likely a linear structure. You cannot do this for a tree or a graph.

---

50. Which command is used to create a new row in SQL ?

(A) Insert into
(B) Alter table
(C) Update
(D) All of these

Correct Answer: (A) Insert into

Solution:

Step 1: Understanding the Concept:
This question asks for the standard SQL command used to add new records (rows) to an existing table.

Step 2: Detailed Explanation:

- **(A) INSERT INTO:** This is the correct Data Manipulation Language (DML) command to add one or more new rows to a table. For example: 'INSERT INTO Customers (Name, City) VALUES ('John Doe', 'New York');'.

- **(B) ALTER TABLE:** This is a Data Definition Language (DDL) command used to modify the structure of the table itself (e.g., add or delete columns), not the data rows.

- **(C) UPDATE:** This is a DML command used to modify data in existing rows, not create new ones.

Step 3: Final Answer:
The ‘INSERT INTO‘ command is used to create a new row in SQL.

Quick Tip

Think of the table’s structure vs. its content. DDL commands (‘CREATE‘, ‘ALTER‘, ‘DROP‘) modify the structure. DML commands (‘SELECT‘, ‘INSERT‘, ‘UPDATE‘, ‘DELETE‘) modify the content (the data rows).

---

51. Which command is used to modify existing data in SQL ?

(A) Select
(B) Update
(C) Delete
(D) None of these

Correct Answer: (B) Update

Solution:

Step 1: Understanding the Concept:
The question asks for the specific SQL command used to change or modify the values within records that already exist in a table.

Step 2: Detailed Explanation:

- **(A) SELECT:** This command is used to retrieve or read data from a table. It does not modify data.

- **(B) UPDATE:** This is the correct DML command used to modify existing rows. The ‘SET‘ clause specifies the new values, and the ‘WHERE‘ clause specifies which rows to modify. Example: ‘UPDATE Customers SET City = ’Boston’ WHERE CustomerID = 1;‘.

- **(C) DELETE:** This command is used to remove entire rows from a table.

Step 3: Final Answer:
The ‘UPDATE‘ command is used to modify existing data in SQL.

Quick Tip

Remember the acronym CRUD for basic data operations:

- **C**reate - ‘INSERT‘
- **R**ead - ‘SELECT‘
- **U**pdate - ‘UPDATE‘
- **D**elete - ‘DELETE‘

---

52. In a Boolean Algebra pVp is equal to

(A) 1
(B) 0
(C) 2
(D) none of these

Correct Answer: (D) none of these

Solution:

Step 1: Understanding the Concept:
The question refers to the Idempotent Law of Boolean algebra. The symbol ‘V‘ represents the logical OR operation (also written as ‘+‘).

Step 2: Key Formula or Approach:
The Idempotent Law states that for any Boolean variable ‘p‘:

$$p + p = p$$

$$p \cdot p = p$$

In the notation of the question, this is $p \vee p = p$.

Step 3: Detailed Explanation:
We can verify this by considering the possible values of ‘p‘:

- If $p = 0$, then $p \vee p = 0 \vee 0 = 0$. So the result is ‘p‘.
- If $p = 1$, then $p \vee p = 1 \vee 1 = 1$. So the result is ‘p‘.

In both cases, the result of $p \vee p$ is ‘p‘ itself.
Now let’s look at the given options: (A) 1, (B) 0, (C) 2. None of these is the variable ‘p‘. The result is dependent on ‘p‘, not a constant value of 0 or 1. Therefore, none of the options A, B,

or C is correct.

Step 4: Final Answer:
Since $p \vee p = p$, and 'p' is not listed as an option, the correct answer is "none of these".

Quick Tip

Don't confuse the Idempotent Law ('p + p = p') with the Complementation Law ('p + p' = 1'). Questions often test your ability to distinguish between these fundamental laws of Boolean algebra.

---

53. A (B + C) = AB + AC ; A + BC = (A + B) (A + C) represent which law?

(A) Commutative
(B) Associative
(C) Distributive
(D) Idempotent

Correct Answer: (C) Distributive

Solution:

Step 1: Understanding the Concept:
The question presents two fundamental identities from Boolean algebra and asks to name the law they represent.

Step 2: Key Formula or Approach:
Let's review the definitions of the laws mentioned:

- **Commutative Law:** Order doesn't matter. $A + B = B + A$ and $AB = BA$.
- **Associative Law:** Grouping doesn't matter. $A + (B + C) = (A + B) + C$ and $A(BC) = (AB)C$.
- **Idempotent Law:** $A + A = A$ and $AA = A$.
- **Distributive Law:** This law describes how two operations interact. The two forms are:
  1. Multiplication distributes over addition: $A(B + C) = AB + AC$
  2. Addition distributes over multiplication: $A + BC = (A + B)(A + C)$

Step 3: Detailed Explanation:
The expressions given in the question, $A(B+C) = AB+AC$ and $A+BC = (A+B)(A+C)$, are the exact definitions of the two forms of the Distributive Law in Boolean algebra.

Step 4: Final Answer:
The given expressions represent the Distributive Law.

Quick Tip

While the first form of the distributive law, ‘A(B+C) = AB+AC‘, is familiar from ordinary algebra, the second form, ‘A+BC = (A+B)(A+C)‘, is unique to Boolean algebra. Memorizing both is key.

---

54. Which keyword is used to define macros in C++ ?

(A) # macro
(B) # define
(C) macro
(D) define

Correct Answer: (B) # define

Solution:

Step 1: Understanding the Concept:
Macros in C and C++ are a feature of the preprocessor. The preprocessor scans the code before the compiler does and performs text substitutions. The question asks for the specific preprocessor directive used to create a macro.

Step 2: Detailed Explanation:

- **(B) # define:** This is the correct preprocessor directive. The ‘#‘ symbol indicates that this is a directive for the preprocessor, and ‘define‘ is the command to create a macro. For example, ‘#define PI 3.14159‘ tells the preprocessor to replace every subsequent occurrence of the token ‘PI‘ with ‘3.14159‘.

- **(A) # macro:** Incorrect syntax.

- **(C) macro, (D) define:** Incorrect. The ‘#‘ prefix is mandatory for preprocessor directives.

Step 3: Final Answer:
The ‘#define‘ directive is used to define macros in C++.

Quick Tip

All preprocessor directives in C/C++ start with a ‘#‘ symbol. Common directives include ‘#include‘, ‘#define‘, ‘#ifdef‘, ‘#endif‘, and ‘#pragma‘.

---

55. Which of the following operators is known as the ’NOT’ operator ?

(A) ˜
(B) !
(C) &&
(D) ——

Correct Answer: (B) !

Solution:

Step 1: Understanding the Concept:
The ’NOT’ operation in programming is a unary operation that inverts the logical value of its operand. The question asks to identify the symbol for this operator in C++.

Step 2: Detailed Explanation:
C++ has two ’NOT’ operators for different purposes:

- **(B) ! (Logical NOT):** This operator is used with boolean values. It converts ‘true‘ to ‘false‘ and ‘false‘ to ‘true‘. This is what is generally referred to as the ’NOT’ operator in the context of logical expressions.

- **(A) ˜(Bitwise NOT):** This operator is used with integer types. It inverts every bit of its operand (0s become 1s and 1s become 0s).

- **(C) && (Logical AND):** This is a binary operator that returns ‘true‘ if both its operands are ‘true‘.

- **(D) —— (Logical OR):** This is a binary operator that returns ‘true‘ if at least one of its operands is ‘true‘.

Given the options, and the common terminology, ’NOT’ refers to the logical NOT operator ‘¡.

Step 3: Final Answer:
The '¡ operator is known as the 'NOT' operator.

Quick Tip

Distinguish between logical operators ('¡, '&&', '——') which work on boolean logic (true/false) and bitwise operators (' ', '&', '—') which work on the individual bits of numbers.

---

56. Index of the array starts from

(A) 0
(B) 1
(C) 2
(D) -1

Correct Answer: (A) 0

Solution:

Step 1: Understanding the Concept:
In most modern programming languages, including C++, Java, and Python, arrays are zero-indexed. This means the numbering of elements begins with 0.

Step 2: Detailed Explanation:
An array is a collection of elements stored in contiguous memory locations. To access a specific element, we use its index.
If an array 'arr' has 'n' elements, the indices of these elements will range from '0' to 'n-1'.

- The first element is at index '0'.
- The second element is at index '1'.
- ...and so on, up to the last (n-th) element, which is at index 'n-1'.

Therefore, the index of an array starts from 0.

Quick Tip

Remembering that arrays are 0-indexed is crucial for avoiding ”off-by-one” errors, which are a common type of bug in programming, especially when writing loops that iterate over arrays.

---

57. Which one of the following is associated with logic gate ?

(A) AND
(B) For
(C) IFU
(D) Break

Correct Answer: (A) AND

Solution:

Step 1: Understanding the Concept:
A logic gate is a basic building block of a digital circuit. It takes one or more binary inputs and produces a single binary output based on a specific logical function. The question asks to identify a term that represents a logic gate.

Step 2: Detailed Explanation:

- **(A) AND:** An AND gate is a fundamental logic gate that outputs 1 only if all of its inputs are 1. This is directly associated with logic gates.

- **(B) For:** ‘For‘ is a keyword in many programming languages used to create a loop. It is a control flow structure, not a hardware logic gate.

- **(C) IFU (Instruction Fetch Unit):** This is a component of a CPU responsible for fetching instructions from memory. It is part of computer architecture, not a logic gate itself (though it is built from them).

- **(D) Break:** ‘Break‘ is a keyword in programming used to exit from a loop or a switch statement. It is a control flow statement.

Step 3: Final Answer:
AND is a type of logic gate.

Quick Tip

Remember the seven basic logic gates: AND, OR, NOT, NAND (NOT-AND), NOR (NOT-OR), XOR (Exclusive-OR), and XNOR (Exclusive-NOR).

---

58. Which one of the following is an operating system ?

(A) LAN
(B) NIC
(C) www
(D) None of these

Correct Answer: (D) None of these

Solution:

Step 1: Understanding the Concept:
An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs. The question asks to identify an OS from the given list.

Step 2: Detailed Explanation:

- **(A) LAN (Local Area Network):** This is a type of computer network that interconnects computers within a limited area. It is a network concept, not an OS.
- **(B) NIC (Network Interface Card):** This is a piece of computer hardware that allows a computer to connect to a network. It is a physical device, not an OS.
- **(C) www (World Wide Web):** This is an information system where documents and other web resources are identified by URLs, interlinked by hypertext links, and can be accessed via the Internet. It is a service, not an OS.

Since none of the options are an operating system, the correct choice is ”None of these”. Examples of operating systems include Microsoft Windows, macOS, Linux, Android, and iOS.

Step 3: Final Answer:
None of the given options is an operating system.

Quick Tip

An operating system is the main software that runs on a computer, without which the computer is useless (e.g., Windows, Linux). Distinguish it from hardware (like a NIC) and network concepts (like LAN or WWW).

---

59. What are the digits used to represent binary numbers ?

(A) 0, 1
(B) 1, 2
(C) 0, 1, 2
(D) 0, 2

Correct Answer: (A) 0, 1

Solution:

Step 1: Understanding the Concept:
The binary number system is a base-2 number system. The ”base” (or radix) of a number system indicates how many unique digits are used to represent numbers.

Step 2: Detailed Explanation:
Since the binary system is base-2, it uses only two distinct digits. These digits are 0 and 1. All binary numbers are constructed using combinations of these two digits.
For example, the decimal number 5 is represented as 101 in binary.

Step 3: Final Answer:
The digits used to represent binary numbers are 0 and 1.

Quick Tip

The name of the number system often reveals its base:

- **Bin**ary: Base 2 (digits 0, 1)
- **Oct**al: Base 8 (digits 0-7)
- **Dec**imal: Base 10 (digits 0-9)
- **Hexa**decimal: Base 16 (digits 0-9, A-F)

---

60. A C++ line ends with which symbol ?

(A) ;
(B) !
(C) ,
(D) :

Correct Answer: (A) ;

Solution:

Step 1: Understanding the Concept:
In C++ and other C-family languages, a specific punctuation mark is required to terminate a statement, signaling to the compiler that the statement is complete.

Step 2: Detailed Explanation:

- **(A) ; (Semicolon):** This is the statement terminator in C++. Every complete instruction or statement (like variable declarations, assignments, function calls) must end with a semicolon. For example: ‘int x = 5;‘.

- **(B) ! (Exclamation mark):** This is the logical NOT operator.

- **(C) , (Comma):** This is used as a separator, for example, to separate function arguments or multiple variable declarations in a single line (‘int a, b;‘).

- **(D) : (Colon):** This is used for various purposes, including defining labels (for ‘goto‘), specifying access control in classes (‘public:‘), and in ternary operators (‘? :‘).

Step 3: Final Answer:
A C++ statement (line of instruction) ends with a semicolon (;).

Quick Tip

A missing semicolon is one of the most frequent syntax errors for beginners in C++. If you get a strange compiler error, one of the first things to check is whether the preceding line is correctly terminated with a semicolon.

---

61. The elements of linked list are stored

(A) in a structure
(B) in an array
(C) anywhere in the disk

(D) in continuous memory location

Correct Answer: (A) in a structure

Solution:

Step 1: Understanding the Concept:
A linked list is a linear data structure where elements are not stored at contiguous memory locations. Instead, each element (called a node) contains the data itself and a pointer (or link) to the next node in the sequence. The question asks how these elements are stored.

Step 2: Detailed Explanation:
Let's analyze the options:

- **(A) in a structure:** Each node of a linked list is typically implemented as a structure (or a class in C++). This structure contains two main parts: the data element and a pointer to the next node. So, the elements are stored *within* these structures. This statement is factually correct about the composition of a linked list's nodes.

- **(B) in an array:** This is incorrect. Storing elements in an array is the basis of an array-based list, which is a different data structure from a linked list.

- **(C) anywhere in the disk:** This is incorrect. A standard linked list is an in-memory data structure; its elements are stored in RAM (main memory), not on the disk.

- **(D) in continuous memory location:** This is the defining characteristic of an array, not a linked list. The main advantage of a linked list is that its nodes can be stored anywhere in non-contiguous memory.

Among the given choices, the most accurate description of how an individual element is packaged and stored is "in a structure" (which acts as the node). While the list as a whole is stored in non-contiguous memory, this option is not available in a correctly worded form.

Step 3: Final Answer:
The elements of a linked list are stored in nodes, which are commonly implemented as structures.

Quick Tip

Remember the key difference between an array and a linked list: An array stores elements contiguously in memory. A linked list stores elements in nodes scattered across memory, connected by pointers.

62. A program statement written in SQL is called

(A) Query
(B) Procedure
(C) Function
(D) Record

Correct Answer: (A) Query

Solution:

Step 1: Understanding the Concept:
SQL (Structured Query Language) is the language used to communicate with databases. A command sent to the database to perform an action is known by a specific term.

Step 2: Detailed Explanation:

- **(A) Query:** This is the general term for a request for data or information from a database. Statements like ‘SELECT‘, ‘INSERT‘, ‘UPDATE‘, and ‘DELETE‘ are all types of queries. It is the most appropriate term for an SQL statement.

- **(B) Procedure (Stored Procedure):** A procedure is a set of one or more SQL statements saved in the database that can be executed as a single unit. A single statement is a query, not a procedure.

- **(C) Function (User-Defined Function):** A function is a named routine that accepts parameters and returns a value. It is a type of database object, but not the general term for a single statement.

- **(D) Record:** A record (also called a row or tuple) is a single entry of data in a table. It is the data itself, not the statement used to access it.

Step 3: Final Answer:
A program statement written in SQL is called a query.

Quick Tip

Think of ”Query” as asking the database a question or giving it a command. ‘SELECT‘ asks ”What data do you have?”, while ‘INSERT‘ commands it to ”Store this data!”.

---

63. Which of the following is an entry control loop ?

(A) For
(B) While
(C) Do-while
(D) None of these

Correct Answer: (A) For (and (B) While)

Solution:

Step 1: Understanding the Concept:
Loops in programming are categorized based on when the loop's controlling condition is checked.

- **Entry-controlled loop:** The condition is checked *before* the loop body is executed. If the condition is false initially, the loop body will not execute even once.

- **Exit-controlled loop:** The condition is checked *after* the loop body is executed. This guarantees that the loop body will execute at least one time.

Step 2: Detailed Explanation:

- **(A) For loop:** The condition in a 'for' loop ('for (init; condition; increment)') is checked before each iteration. Thus, it is an entry-controlled loop.

- **(B) While loop:** The condition in a 'while' loop ('while (condition)') is checked before the loop body is executed. Thus, it is also an entry-controlled loop.

- **(C) Do-while loop:** The condition in a 'do-while' loop ('do { ... } while (condition);') is checked at the end of the loop body. Thus, it is an exit-controlled loop.

Since both 'for' and 'while' are entry-controlled loops, options (A) and (B) are both correct. In a single-choice format, this indicates a potentially flawed question, but both are valid examples. We select (A) as a correct representative.

Step 3: Final Answer:
The 'For' loop is an entry-controlled loop, as is the 'While' loop.

Quick Tip

Remember: 'while' and 'for' might run zero times. 'do-while' always runs at least once. This is the key difference between entry-controlled and exit-controlled loops.

64. C++ language is a successor to which language ?

(A) B
(B) C
(C) Java
(D) VB

Correct Answer: (B) C

Solution:

Step 1: Understanding the Concept:
This question pertains to the history of programming languages and the evolution of C++.

Step 2: Detailed Explanation:

- The C++ programming language was developed by Bjarne Stroustrup at Bell Labs starting in 1979.
- It was designed as an extension of the **C** language. The initial name was "C with Classes," reflecting its addition of object-oriented features (like classes) to the existing C language syntax and semantics.
- The name was later changed to C++, with the '++' being the C language's increment operator, as a pun to suggest it is an incremented or improved version of C.
- The B language was the predecessor to C. Java and VB (Visual Basic) were developed later and are not direct predecessors of C++.

Step 3: Final Answer:
C++ is a successor to the C language.

Quick Tip

The lineage of these influential languages is: B → C → C++. Each language built upon the ideas of its predecessor.

---

65. How many keywords are there in C++ ?

(A) 82
(B) 48

(C) 99
(D) 95

Correct Answer: (D) 95

Solution:

Step 1: Understanding the Concept:
Keywords (or reserved words) are words that have a special meaning to the C++ compiler and cannot be used as identifiers (like variable or function names). The number of keywords is defined by the C++ language standard and has increased over time with new versions.

Step 2: Detailed Explanation:
The number of keywords in C++ has evolved with each new standard:

- C++98/03 had 63 keywords.
- C++11 added 10 new keywords (e.g., ‘auto‘, ‘nullptr‘, ‘override‘).
- C++17 added more keywords.
- C++20 and C++23 have added even more keywords (e.g., ‘concept‘, ‘requires‘, ‘consteval‘).

As of the C++23 standard, the total number of keywords is around 95. Given the options, 95 is the most accurate count for a recent version of the C++ standard.

Step 3: Final Answer:
There are approximately 95 keywords in modern C++.

Quick Tip

You don’t need to memorize the exact number of keywords for every standard. Just be aware that the number is significant (much more than C’s 32) and has been growing, so a number in the 90s is a reasonable guess for a modern C++ exam question.

---

66. Which of the following operators is known as scope resolution operator ?

(A) ::
(B) ?:
(C) -¿

(D) ;

Correct Answer: (A) ::

Solution:

Step 1: Understanding the Concept:
The scope resolution operator is a unique C++ operator used to define a member of a class outside the class itself or to access a global variable when there is a local variable with the same name.

Step 2: Detailed Explanation:

- **(A) :: (Scope Resolution Operator):** This is the correct operator. It is used to specify a scope. For example, ‘std::cout‘ uses ‘::‘ to access ‘cout‘ from the ‘std‘ namespace. Another use is ‘ClassName::member‘ to access static members or define member functions outside the class.

- **(B) ?:** This is the ternary conditional operator.

- **(C) ->:** This is the arrow operator, used to access a member of a structure or class through a pointer.

- **(D) ;:** This is the semicolon, used as a statement terminator.

Step 3: Final Answer:
The ‘::‘ operator is known as the scope resolution operator.

Quick Tip

Think of ‘::‘ as saying ”from the scope of”. For ‘std::cout‘, it means ”the ‘cout‘ that belongs to the ‘std‘ scope”.

---

67. How many types of loop are there in C++ ?

(A) 2
(B) 3
(C) 4
(D) 1

Correct Answer: (B) 3

Solution:

Step 1: Understanding the Concept:
Loops are control flow structures in C++ that allow a block of code to be executed repeatedly. The question asks for the number of fundamental loop types available in the language.

Step 2: Detailed Explanation:
C++ provides three primary types of looping constructs:

1. **'for' loop:** Typically used when the number of iterations is known beforehand. It consists of an initialization, a condition, and an increment/decrement part.

2. **'while' loop:** An entry-controlled loop that executes as long as a specified condition is true. The condition is checked before each iteration.

3. **'do-while' loop:** An exit-controlled loop that is similar to a 'while' loop, but the condition is checked after the loop body executes, guaranteeing at least one execution.

Modern C++ also introduced the range-based 'for' loop, but it is considered a variation of the 'for' loop, not a fundamentally different type. Therefore, the standard answer is 3.

Step 3: Final Answer:
There are 3 fundamental types of loops in C++.

Quick Tip

Remember the three main loops: 'for', 'while', and 'do-while'. Choose 'for' for counted loops, 'while' for conditional loops that might not run at all, and 'do-while' for conditional loops that must run at least once.

---

68. Which one is an input function for stack ?

(A) POP
(B) PUSH
(C) Binary
(D) Traversal

Correct Answer: (B) PUSH

Solution:

Step 1: Understanding the Concept:
A stack is a data structure that follows the Last-In, First-Out (LIFO) principle. It has standard operations for adding and removing elements. The question asks for the name of the operation used to add (input) an element.

Step 2: Detailed Explanation:
The two primary operations for a stack are:

- **PUSH:** Adds an element to the top of the stack. This is the ”input” operation.

- **POP:** Removes the element from the top of the stack. This is the ”output” or removal operation.

The other terms are not standard stack operations. ”Traversal” refers to visiting all elements of a data structure, and ”Binary” is not an operation.

Step 3: Final Answer:
The ‘PUSH‘ function is the input function for a stack.

Quick Tip

Think of a stack of plates. You ‘PUSH‘ a new plate onto the top of the stack, and you ‘POP‘ a plate off the top to use it.

---

69. Which one is a DBMS language ?

(A) C
(B) SQL
(C) Java
(D) C++

Correct Answer: (B) SQL

Solution:

Step 1: Understanding the Concept:
A DBMS (Database Management System) language is a specialized language used to define, manipulate, and query data in a database. The question asks to identify such a language.

Step 2: Detailed Explanation:

- **(B) SQL (Structured Query Language):** This is the standard language specifically designed for interacting with relational databases. It is used to create tables, insert data, update data, delete data, and retrieve data. It is the quintessential DBMS language.

- **(A) C, (C) Java, (D) C++:** These are general-purpose programming languages. They are used to build applications, and these applications can use libraries (like JDBC in Java or ODBC in C++) to connect to and interact with a database, but they are not themselves DBMS languages.

Step 3: Final Answer:
SQL is a DBMS language.

Quick Tip

SQL is the language of databases. General-purpose languages like Java, C++, Python, etc., are the languages of applications that *use* databases.

---

70. Base of Octal number system is

(A) 10
(B) 8
(C) 2
(D) 16

Correct Answer: (B) 8

Solution:

Step 1: Understanding the Concept:
The base, or radix, of a number system is the number of unique digits or symbols used to represent numbers in that system. The name of the system often indicates its base.

Step 2: Detailed Explanation:

- The **Octal** number system gets its name from the Latin root ”octo,” meaning eight.

- It uses eight distinct digits: 0, 1, 2, 3, 4, 5, 6, and 7.

- Therefore, the base of the octal number system is 8.

- For reference: Base 10 is Decimal, Base 2 is Binary, and Base 16 is Hexadecimal.

Step 3: Final Answer:
The base of the Octal number system is 8.

Quick Tip

Pay attention to the prefixes: ‘Bi‘ for 2, ‘Oct‘ for 8, ‘Dec‘ for 10, and ‘Hex‘ for 16. This makes it easy to remember the base of each common number system.

Section - B

1. What is constructor in C++ ?

Solution:

Step 1: Definition
A constructor in C++ is a special member function of a class that is automatically called when an object of that class is created. Its primary purpose is to initialize the data members of the new object.

Step 2: Key Characteristics

- It has the same name as the class.
- It does not have a return type, not even ‘void‘.
- It is called automatically by the compiler when an object is instantiated.
- If no constructor is defined by the user, the compiler provides a default constructor.

Step 3: Types of Constructors

1. **Default Constructor:** A constructor that takes no arguments.
2. **Parameterized Constructor:** A constructor that accepts one or more arguments to initialize the object.

3. **Copy Constructor:** A constructor that creates an object by initializing it with an existing object of the same class.

Step 4: Example

```
class Car {
public:
    int speed;

    // Parameterized Constructor
    Car(int s) {
        speed = s;
    }
};

int main() {
    // Constructor is called when carObj is created
    Car carObj(100);
    return 0;
}
```

Quick Tip

Remember that a constructor's job is to *initialize* an object. Using a member initializer list (e.g., ‘Car(int s) : speed(s) ‘) is often more efficient than assigning values inside the constructor body, as it performs initialization directly rather than default initialization followed by assignment.

---

2. Is it possible to overload a destructor? Give reason.

Solution:

Step 1: Direct Answer

No, it is not possible to overload a destructor in C++.

Step 2: Reason

Function overloading requires that functions have the same name but different parameter lists (either different types or a different number of arguments).

- A destructor in C++ has a very specific, fixed signature: it must have the same name as the class, prefixed with a tilde (‘ ‘), and it cannot take any arguments. For a class named ‘MyClass‘, the destructor is always ‘ MyClass()‘.

- Since a destructor cannot have any parameters, there is no way to create another version of it with a different parameter list. Without the ability to change the parameters, overloading is impossible.

Step 3: Logical Explanation
The purpose of a destructor is to clean up resources used by an object just before it is destroyed. This is a singular, unambiguous event. There is only one way for an object to be destroyed, so there is no need for multiple versions of a destructor. The compiler knows exactly which destructor to call (‘ ClassName()‘) without needing arguments to differentiate.

Quick Tip

While you cannot overload a destructor, you *can* and frequently *should* declare a destructor as ‘virtual‘ in a base class if you intend to use polymorphism. A virtual destructor ensures that the correct derived class destructor is called when deleting an object through a base class pointer.

---

3. What is static variable ?

Solution:

Step 1: Definition
A ‘static‘ variable is a variable that has been allocated ”statically,” meaning its lifetime (the period during which it exists in memory) is the entire duration of the program’s execution. Its meaning changes slightly based on its scope.

Step 2: Static Variables in Different Scopes

1. **Static Variable inside a Function (Static Local Variable):**
   A variable declared as ‘static‘ inside a function retains its value between multiple calls to that function. It is initialized only once, the first time the function is called, and it persists until the program ends.
   Example:

   ```
   void counter() {
       static int count = 0; // Initialized only once
       count++;
       // count retains its value across calls
   }
   ```

2. **Static Variable inside a Class (Static Data Member):**
   A data member of a class declared as ‘static‘ is shared among all objects (instances) of

that class. There is only one copy of the static variable for the entire class, regardless of how many objects are created. It is also known as a class variable.
Example:

```
class Player {
public:
    static int playerCount; // Shared by all Player objects
    Player() { playerCount++; }
};
int Player::playerCount = 0; // Initialization
```

Quick Tip

Remember the key properties:

- **Lifetime:** The entire program's execution.
- **Scope:** Can be local (in a function) or class-wide.
- **Key idea:** Persistence (in functions) and Sharing (in classes).

---

4. What is a table ?

Solution:

Step 1: Definition
In the context of a relational database, a table is a data structure that organizes information into a two-dimensional grid of rows and columns. It is the fundamental object for storing a collection of related data.

Step 2: Structure of a Table

- **Columns (or Fields/Attributes):** Each column represents a specific attribute or property of the data. It has a name and a data type (e.g., integer, string, date). For example, in a 'Students' table, columns might be 'StudentID', 'FirstName', and 'Major'.

- **Rows (or Records/Tuples):** Each row represents a single, complete record or instance of the entity the table describes. For example, a single row in the 'Students' table would contain all the information for one specific student.

- **Cell:** The intersection of a row and a column, which contains a single data value.

Step 3: Purpose
The purpose of a table is to store data about a specific entity (like customers, products, or orders) in a structured and organized manner, which allows for efficient querying, manipulation, and management of the data.

> **Quick Tip**
>
> An easy way to visualize a database table is to think of a spreadsheet. The entire sheet is the table, the column headers are the fields, and each row of data is a record.

---

5. What is protocol ?

Solution:

Step 1: Definition
In computing, a protocol is a set of formal rules and conventions that governs how data is formatted, transmitted, and received between computing devices in a network. It ensures that communication is orderly, efficient, and error-free.

Step 2: Key Functions of a Protocol
Protocols define many aspects of communication, including:

- **Data Formatting:** The structure of the data packets being exchanged.
- **Addressing:** How to identify the sender and receiver on the network (e.g., IP addresses).
- **Flow Control:** Managing the rate of data transmission to prevent the receiver from being overwhelmed.
- **Error Control:** Detecting and often correcting errors that occur during transmission (e.g., using checksums).
- **Connection Management:** The sequence of messages to establish and terminate a connection.

Step 3: Examples

- **HTTP (Hypertext Transfer Protocol):** The protocol for the World Wide Web.

- **TCP/IP (Transmission Control Protocol/Internet Protocol):** The fundamental suite of protocols for the Internet.

- **SMTP (Simple Mail Transfer Protocol):** The protocol for sending email.

> Quick Tip
>
> Think of a protocol as the ”language and grammar” that two computers agree to use when they communicate. Without a shared protocol, the data sent by one computer would be meaningless noise to the other.

---

6. Describe the use of linked list.

Solution:

Step 1: Introduction
A linked list is a dynamic data structure where elements (nodes) are linked together using pointers. Its primary advantage over an array is its dynamic size and efficient insertion/deletion operations. This makes it suitable for a variety of applications.

Step 2: Key Uses of Linked Lists

1. **Implementing other Data Structures:** Linked lists are often used as the underlying structure to implement other abstract data types.

   - **Stacks:** A stack can be easily implemented by performing insertions (push) and deletions (pop) at the head of a linked list.

   - **Queues:** A queue can be implemented by performing insertions (enqueue) at the tail and deletions (dequeue) at the head of a linked list.

2. **Dynamic Memory Management:** They are used in situations where the number of elements is not known beforehand and can change frequently. Since nodes are allocated memory dynamically, the list can grow or shrink at runtime as needed.

3. **Applications with Frequent Insertions and Deletions:** Operations like inserting or deleting an element in the middle of a list are much more efficient in a linked list than in an array, as no shifting of elements is required. A classic example is managing a list of tasks in an operating system’s scheduler.

4. **Real-World Applications:**

   - **Music Players:** A playlist where you can easily add, remove, and reorder songs can be implemented using a doubly linked list.

   - **Web Browsers:** The "back" and "forward" button functionality can be implemented using a doubly linked list to navigate through recently visited URLs.

   - **Undo Functionality:** In text editors, a linked list can store a history of operations to allow for "undo" actions.

Quick Tip

The main trade-off to remember is:

- Choose a **linked list** for dynamic size and frequent insertions/deletions.
- Choose an **array** for fast random access (accessing elements by index) and when the size is fixed.

---

7. Define inheritance.

Solution:

Step 1: Definition
Inheritance is one of the fundamental principles of Object-Oriented Programming (OOP). It is a mechanism in which a new class (known as the **derived** or **child** class) acquires the properties (data members) and behaviors (member functions) from an existing class (known as the **base** or **parent** class).

Step 2: The "Is-A" Relationship
Inheritance is used to model an "is-a" relationship between classes. For example, a ‘Car‘ "is-a" ‘Vehicle‘. The ‘Car‘ class can inherit common attributes like ‘speed‘ and ‘color‘ from the ‘Vehicle‘ class, and then add its own specific attributes like ‘numberOfDoors‘.

Step 3: Purpose and Benefits
The primary benefits of using inheritance are:

- **Code Reusability:** It allows us to reuse the code of an existing class. The derived class inherits all the functionality of the base class without having to rewrite it.

- **Extensibility:** We can add new features to the derived class without modifying the base class.
- **Polymorphism:** Inheritance is a prerequisite for polymorphism, which allows an object of a derived class to be treated as an object of its base class.

Quick Tip

Use real-world analogies to remember the concept. A ‘Dog‘ inherits traits from the ‘Mammal‘ class, which in turn inherits from the ‘Animal‘ class. This creates a logical hierarchy and promotes code reuse.

---

8. Define Local Area Network.

Solution:

Step 1: Definition
A Local Area Network (LAN) is a computer network that interconnects computers and other network devices within a limited geographical area.

Step 2: Scope and Scale
The area covered by a LAN is typically small, such as:

- A single home or office.
- One or more floors in an office building.
- A school or university campus.

Step 3: Key Characteristics

- **Private Ownership:** LANs are typically owned, controlled, and managed by a single person or organization.
- **High Speed:** They provide high data-transfer rates (e.g., 100 Mbps, 1 Gbps, or higher).
- **Low Latency:** The time it takes for data to travel between devices is very short due to the small distances.

- **Resource Sharing:** The primary purpose of a LAN is to allow connected devices to share resources like files, printers, applications, and a common internet connection.

Step 4: Common Technologies
The two most common technologies used to build LANs are **Ethernet** (for wired connections) and **Wi-Fi** (for wireless connections).

Quick Tip

To remember the different network types, think of their scope:

- **PAN (Personal):** Your personal bubble (Bluetooth).
- **LAN (Local):** Your building or campus.
- **MAN (Metropolitan):** Your city.
- **WAN (Wide):** The world (The Internet).

9. What is logic gate ?

Solution:

Step 1: Definition
A logic gate is a fundamental building block of a digital circuit. It is an electronic device that performs a basic logical operation on one or more binary inputs to produce a single binary output.

Step 2: How It Works
Logic gates operate on binary values, typically represented as ‘0‘ (False/Off) and ‘1‘ (True/On). The relationship between the input(s) and the output is based on a specific type of Boolean logic. This relationship can be described using a truth table.

Step 3: Basic Types of Logic Gates
The three most fundamental logic gates are:

- **AND Gate:** The output is ‘1‘ only if *all* of its inputs are ‘1‘.

- **OR Gate:** The output is ‘1‘ if *at least one* of its inputs is ‘1‘.

- **NOT Gate (Inverter):** The output is the opposite of its single input. If the input is ‘1‘, the output is ‘0‘, and vice versa.

Other important gates include NAND, NOR, XOR, and XNOR, which can be constructed from the basic three.

Quick Tip

Think of a logic gate as a simple decision-maker. An AND gate is like a security door that only opens if two separate keys (both inputs) are turned simultaneously.

---

10. What is database ?

Solution:

Step 1: Definition
A database is an organized, structured collection of data that is stored electronically in a computer system. The data is managed in such a way that it can be easily accessed, managed, updated, and retrieved.

Step 2: Key Characteristics

- **Structured Data:** Data in a database is organized, often in tables (in the case of relational databases), which makes it efficient to query.
- **Data Integrity:** Databases enforce rules (constraints) to maintain the accuracy and consistency of data.
- **Data Independence:** The way data is stored can be changed without affecting the applications that access it.
- **Concurrent Access:** Databases allow multiple users to access and modify the data simultaneously without interfering with each other.

Step 3: Database vs. DBMS
It is important to distinguish between a database and a Database Management System (DBMS).

- The **database** is the collection of data itself.
- The **DBMS** is the software used to create, manage, and interact with the database (e.g., MySQL, Oracle, SQL Server).

Quick Tip

A simple analogy: a library is a database (an organized collection of books). The card catalog and the librarians who help you find books are the DBMS (the system that lets you manage and access the collection).

---

11. What is abstract class ?

Solution:

Step 1: Definition
In Object-Oriented Programming, an abstract class is a class that is designed to be a blueprint for other classes. It cannot be used to create objects (it cannot be instantiated). Its main purpose is to be inherited by other classes (concrete classes).

Step 2: Key Features

- **Cannot be Instantiated:** You cannot create an object of an abstract class. For example, 'AbstractClass obj;' would result in a compilation error.

- **Contains Abstract Methods:** An abstract class typically contains one or more **abstract methods** (in C++, these are called **pure virtual functions**). An abstract method is a method that is declared but contains no implementation.

- **Forces Implementation:** Any concrete subclass that inherits from an abstract class must provide an implementation for all of its abstract methods. This enforces a common interface among all subclasses.

- **Can have Concrete Methods:** An abstract class can also contain regular (concrete) methods with implementations and data members, which are inherited by the subclasses.

Step 3: Example in C++

```
class Shape { // Abstract Base Class
public:
    // Pure virtual function (abstract method)
    virtual void draw() = 0;
};

class Circle : public Shape { // Concrete Class
public:
    void draw() override { // Provides implementation
        // code to draw a circle
    }
};
```

Quick Tip

Think of an abstract class as an "incomplete idea." The ‘Shape‘ class is an abstract concept; you can’t create a generic "shape." You can only create concrete shapes like ‘Circle‘ or ‘Square‘, which must provide their own specific way to be drawn.

---

12. Explain the rule of association in Boolean Algebra.

Solution:

Step 1: Definition

The rule of association, or the **Associative Law**, is a fundamental law in Boolean algebra. It states that when an operation (either AND or OR) is applied to three or more variables, the order in which the operations are grouped does not affect the final result, as long as the operator is the same throughout.

Step 2: Forms of the Associative Law

The law has two forms, one for the logical OR operation (+) and one for the logical AND operation (·).

1. **Associative Law of Addition (OR):**
   This states that the ORing of several variables will produce the same result regardless of the grouping.

$$A + (B + C) = (A + B) + C$$

2. **Associative Law of Multiplication (AND):**
   This states that the ANDing of several variables will produce the same result regardless of the grouping.

$$A \cdot (B \cdot C) = (A \cdot B) \cdot C$$

Step 3: Significance

This law allows us to remove parentheses in expressions that contain only AND operators or only OR operators, simplifying the expression. For instance, $A+(B+C)$ can be written simply as $A + B + C$.

Quick Tip

Do not confuse the Associative Law with the Distributive Law. The Associative Law applies only when *all* operators in the expression are the same (e.g., all ORs). The Distributive Law, ‘A(B+C) = AB+AC‘, applies when there is a mix of AND and OR operators.

---

13. What is transmission mode ?

Solution:

Step 1: Definition
Transmission mode, also known as communication mode, refers to the mechanism of transferring data between two devices in a network. It defines the direction of signal flow.

Step 2: Types of Transmission Modes
There are three main types of transmission modes:

1. **Simplex Mode:**
   Communication is unidirectional, like a one-way street. Data flows in only one direction. One device can only transmit, and the other can only receive.
   Example: A television broadcast, where the signal travels from the transmitter to your TV, but not back. A keyboard sending input to a computer is another example.

2. **Half-Duplex Mode:**
   Each device can both transmit and receive, but *not at the same time*. The communication is two-way, but only one device can send data at any given moment.
   Example: A walkie-talkie. While one person is speaking, the other must listen, and they take turns transmitting.

3. **Full-Duplex Mode:**
   Both devices can transmit and receive *simultaneously*. The communication is two-way and concurrent.
   Example: A telephone conversation, where both people can speak and hear at the same time. Modern network connections are typically full-duplex.

Quick Tip

Use road analogies to remember the modes:

- **Simplex:** A one-way street.
- **Half-Duplex:** A single-lane bridge where cars must take turns crossing.
- **Full-Duplex:** A two-lane highway with traffic flowing in both directions at once.

---

14. What is array ?

Solution:

Step 1: Definition
An array is a fundamental, linear data structure that stores a collection of elements.

Step 2: Key Characteristics

- **Homogeneous Elements:** All elements in an array must be of the same data type (e.g., an array of integers, an array of characters).
- **Contiguous Memory:** The elements of an array are stored in adjacent, or contiguous, memory locations. This allows for efficient calculation of an element's address.
- **Fixed Size:** In many languages (like C++ and Java), a simple array has a fixed size that is determined when it is created and cannot be changed later.
- **Indexed Access:** Each element in the array has a unique index, which is a numerical key used to access it. Indexing is typically zero-based, meaning the first element is at index 0, the second at index 1, and so on.

Step 3: Advantages and Disadvantages

- **Advantage:** Fast random access. Because the memory is contiguous, the location of any element can be calculated directly from its index in constant time, O(1).
- **Disadvantage:** Inefficient insertion and deletion. Adding or removing an element from the middle of an array requires shifting all subsequent elements, which is a slow operation, O(n).

Quick Tip

Think of an array like a row of numbered mailboxes. Each mailbox (element) has a unique number (index), holds the same type of item (data type), and they are all lined up next to each other (contiguous memory).

---

15. Explain Data Anomalies.

Solution:

Step 1: Definition
Data anomalies are problems or inconsistencies that can occur in poorly designed databases, typically due to data redundancy (unnecessary repetition of data). These anomalies can lead to data integrity issues when data is inserted, updated, or deleted.

Step 2: Types of Data Anomalies
There are three main types of data anomalies:

1. **Insertion Anomaly:**
   This occurs when you are unable to insert a new piece of information into the database because another, required piece of information is missing.
   Example: Consider a table that stores student and course information together. If you want to add a new course that currently has no students enrolled, you cannot insert it into the table if the 'StudentID' field cannot be null.

2. **Deletion Anomaly:**
   This occurs when the deletion of one piece of data unintentionally causes the loss of other, unrelated data.
   Example: In the same student-course table, if a student is enrolled in only one course, and you delete that enrollment record, you might accidentally delete all information about that student from the database.

3. **Update (or Modification) Anomaly:**
   This occurs when updating a single piece of data requires it to be changed in multiple places. If all occurrences are not updated consistently, the database will contain conflicting information.
   Example: If a student's address is stored in every course enrollment record, changing the student's address would require updating multiple rows. Missing even one update would lead to an inconsistent state.

Step 3: Solution
Data anomalies are prevented through a process called **normalization**, which involves orga-

nizing data into multiple related tables to minimize redundancy.

Quick Tip

The root cause of all data anomalies is **data redundancy**. The solution is always **normalization**. Remember this cause-and-effect relationship.

---

16. Describe search engine.

Solution:

Step 1: Definition
A search engine is a complex software system designed to carry out searches on the World Wide Web. It enables users to find information by entering keywords or phrases into a search bar. The search engine then returns a list of relevant results, typically web pages, images, videos, or other types of files.

Step 2: Core Components and Processes
A search engine works in three primary stages:

1. **Crawling (or Web Crawling/Spidering):**
   The search engine uses automated programs called "crawlers" or "spiders" to systematically browse the internet. These crawlers follow hyperlinks from one web page to another to discover new and updated content.

2. **Indexing:**
   Once a page is crawled, its content is analyzed and stored in a massive database called an "index". The index is like a giant library catalog for the web. It stores information about the words on pages, where they are located, the links between pages, and other metadata. This allows for extremely fast retrieval of information.

3. **Searching and Ranking:**
   When a user enters a query, the search engine scans its index for matching pages. It then uses a complex algorithm to rank the retrieved pages based on their relevance and authority. Factors in ranking can include the presence of keywords, the number and quality of links pointing to the page (e.g., Google's PageRank algorithm), user location, and many other signals. The ranked results are then presented to the user.

Step 3: Examples
The most well-known search engines include Google, Bing, and DuckDuckGo.

Quick Tip

Remember the three-step process: **Crawl** (discover the content), **Index** (organize the content), and **Rank** (deliver the best results). This simple model is the foundation of how all major search engines operate.

---

17. Explain stack.

Solution:

Step 1: Definition
A stack is an abstract data type and a linear data structure that serves as a collection of elements. It is characterized by its access principle.

Step 2: The LIFO Principle
A stack operates on the **Last-In, First-Out (LIFO)** principle. This means that the last element added to the stack will be the first element to be removed. A common real-world analogy is a stack of plates: you place a new plate on top, and you remove a plate from the top.

Step 3: Principal Operations
The two main operations performed on a stack are:

- **Push:** This operation adds a new element to the ”top” of the stack.

- **Pop:** This operation removes the element from the ”top” of the stack.

Other common operations include ‘Peek‘ or ‘Top‘ (returns the top element without removing it) and ‘isEmpty‘ (checks if the stack is empty).

Step 4: Applications
Stacks are used in many areas of computing, including:

- **Function Calls:** The ”call stack” manages active function calls, their local variables, and return addresses.

- **Expression Evaluation:** Converting infix expressions to postfix and evaluating postfix expressions.

- **Undo/Redo Functionality:** In applications like text editors.

- **Backtracking Algorithms:** For example, in solving mazes or in parsing.

Quick Tip

The key to understanding stacks is the LIFO principle. Always visualize a stack of physical objects like books or plates to remember that all operations happen at one end only: the top.

---

18. Define DML.

Solution:

Step 1: Definition
DML stands for **Data Manipulation Language**. It is a subset of SQL (Structured Query Language) used to perform operations on the data stored within the database. Its primary focus is on the manipulation of the data itself, not the structure of the database.

Step 2: Core DML Commands
The four primary DML commands are:

- **SELECT:** Used to retrieve or query data from one or more tables.
  Example: ‘SELECT * FROM Customers;‘

- **INSERT:** Used to add new rows of data into a table.
  Example: ‘INSERT INTO Customers (Name, City) VALUES (’John’, ’New York’);‘

- **UPDATE:** Used to modify existing data within a table.
  Example: ‘UPDATE Customers SET City = ’Boston’ WHERE Name = ’John’;‘

- **DELETE:** Used to remove existing rows from a table.
  Example: ‘DELETE FROM Customers WHERE Name = ’John’;‘

Quick Tip

Contrast DML with DDL (Data Definition Language).

- **DML** manipulates the **data** (the rows inside the tables).
- **DDL** defines the **structure** (the tables themselves, columns, constraints). DDL commands include ‘CREATE‘, ‘ALTER‘, and ‘DROP‘.

---

19. What is a primary key ?

Solution:

Step 1: Definition
A primary key is a constraint in a relational database table that uniquely identifies each record (row) in that table. It is a specific column or a set of columns chosen to be the main identifier for the table's records.

Step 2: Core Properties
A primary key must satisfy two essential rules:

1. **Uniqueness:** The value of the primary key must be unique for each row in the table. No two rows can have the same primary key value.

2. **Non-Null:** The primary key column(s) cannot contain NULL values. Every record must have a value for its primary key.

Step 3: Purpose
The primary key serves several crucial purposes:

- **Enforces Data Integrity:** By ensuring uniqueness, it prevents duplicate records from being entered into the table.

- **Unique Identification:** It provides a reliable way to locate and reference a specific row.

- **Establishing Relationships:** It is used by foreign keys in other tables to create links and relationships between tables.

Example: In a 'Students' table, the 'StudentID' column would be an excellent candidate for a primary key, as each student has a unique ID number.

Quick Tip

Think of a primary key as the Social Security Number (or national ID) for a database record. It's unique, it's never missing, and it's the most reliable way to identify that specific record.

---

20. What is topology ?

Solution:

Step 1: Definition
In the context of computer networks, **topology** refers to the arrangement or layout of the elements (such as nodes, links, and other devices) of a communication network.

Step 2: Physical vs. Logical Topology
Topology can be described in two ways:

- **Physical Topology:** This describes the actual physical layout of the network's components, including the cables and devices. It's how the network looks.
- **Logical Topology:** This describes the path that the data signals take through the network from one device to another, regardless of the physical connections. It's how the network behaves.

Step 3: Common Physical Topologies
Some of the most common physical network topologies include:

- **Bus Topology:** All devices are connected to a single central cable, called the bus.
- **Star Topology:** All devices are connected to a central hub or switch. This is the most common topology for modern LANs.
- **Ring Topology:** Devices are connected in a circular loop, with data passing from one device to the next.
- **Mesh Topology:** Every device is connected to every other device (full mesh) or to several other devices (partial mesh), providing high redundancy.
- **Tree Topology:** A hybrid topology that combines characteristics of bus and star topologies.

Quick Tip

Visualize the names to remember the shapes:

- **Star:** A central point with spokes radiating outwards.
- **Ring:** A circle.
- **Bus:** A single straight line.

21. Describe the types of Data structure.

Solution:

Step 1: Introduction
A data structure is a specialized format for organizing, processing, retrieving, and storing data. It defines the relationship between data and the operations that can be performed on it. Data structures can be broadly classified into two main types: Linear and Non-Linear.

Step 2: Linear Data Structures
In linear data structures, elements are arranged in a sequential or linear order, where each element is attached to its previous and next adjacent elements.

- **Array:** A collection of homogeneous elements stored in contiguous memory locations. It allows for fast random access using an index.

- **Linked List:** A collection of elements (nodes) where each node contains data and a pointer to the next node. They are dynamic and allow for efficient insertions/deletions.

- **Stack:** A linear structure that follows the Last-In, First-Out (LIFO) principle. Operations are performed at one end called the ”top”.

- **Queue:** A linear structure that follows the First-In, First-Out (FIFO) principle. Elements are added at the ”rear” and removed from the ”front”.

Step 3: Non-Linear Data Structures
In non-linear data structures, elements are arranged in a hierarchical or networked manner, where an element can be connected to multiple other elements.

- **Tree:** A hierarchical structure consisting of a root node and subtrees of children with a parent-child relationship. A binary tree is a common example.

- **Graph:** A collection of vertices (nodes) and edges that connect pairs of vertices. They are used to represent networks and complex relationships.

Quick Tip

The easiest way to differentiate is to ask: ”Can I arrange all the elements in a single line without losing their relationship?” If yes, it’s linear (like a list). If no, it’s non-linear (like a family tree or a road map).

22. Differentiate between stack and queue data structure.

Solution:

The key differences between a stack and a queue are as follows:

Basis	Stack	Queue
Operating Principle	Follows the **LIFO** (Last-In, First-Out) principle. The last item added is the first one to be removed.	Follows the **FIFO** (First-In, First-Out) principle. The first item added is the first one to be removed.
Structure	Elements are added and removed from the same end, called the "top".	Elements are added at one end (rear) and removed from the other end (front).
Operations	The main operations are **Push** (to insert) and **Pop** (to remove).	The main operations are **Enqueue** (to insert) and **Dequeue** (to remove).
Pointer(s)	Uses a single pointer called the "top" to keep track of the last element.	Uses two pointers: "front" (to track the first element) and "rear" (to track the last element).
Real-world Analogy	A stack of plates or books.	A line of people waiting at a ticket counter.
Applications	Function calls (call stack), expression evaluation, undo/redo features.	CPU scheduling, print job scheduling, network data transfer (buffers).

Quick Tip

To easily remember the principles:

- **Stack (LIFO):** Think of a PEZ dispenser. The last candy you put in is the first one you get out.
- **Queue (FIFO):** Think of a queue for a bus. The first person to get in line is the first person to get on the bus.

23. Explain the different types of SQL commands.

Solution:

SQL (Structured Query Language) commands are categorized into several types based on their function. The main categories are:

1. **DDL (Data Definition Language):**
   These commands are used to define and manage the structure of the database and its objects. They deal with the database schema.

   - **CREATE:** To create new objects like tables, views, or databases.

   - **ALTER:** To modify the structure of an existing database object.

   - **DROP:** To permanently delete an existing object.

   - **TRUNCATE:** To remove all records from a table, including all spaces allocated for the records.

2. **DML (Data Manipulation Language):**
   These commands are used to manipulate the data within the tables.

   - **SELECT:** To retrieve data from the database.

   - **INSERT:** To add new data into a table.

   - **UPDATE:** To modify existing data in a table.

   - **DELETE:** To remove data from a table.

3. **DCL (Data Control Language):**
   These commands are used to manage user permissions and access control to the database.

   - **GRANT:** To give a user access privileges.

   - **REVOKE:** To take back permissions from a user.

4. **TCL (Transaction Control Language):**
   These commands are used to manage transactions in the database, ensuring that groups of operations are completed together or not at all.

   - **COMMIT:** To save a transaction permanently.

   - **ROLLBACK:** To undo a transaction.

   - **SAVEPOINT:** To set a point within a transaction to which you can later roll back.

Quick Tip

Remember the main categories by their purpose:

- **DDL** → Deals with the **Definition** (structure).
- **DML** → Deals with the **Manipulation** (data).
- **DCL** → Deals with the **Control** (permissions).
- **TCL** → Deals with the **Transactions**.

---

24. What is the difference between delete and truncate commands ?

Solution:

The ‘DELETE‘ and ‘TRUNCATE‘ commands in SQL are both used to remove records from a table, but they work differently and have different implications.

Basis	**DELETE**	**TRUNCATE**
Command Type	It is a **DML** (Data Manipulation Language) command.	It is a **DDL** (Data Definition Language) command.
Filtering	Can remove specific rows using a **WHERE** clause. If no WHERE clause is used, it removes all rows.	Removes **all** rows from a table. It cannot be used with a WHERE clause.
Speed	Slower, because it removes rows one by one and logs each deletion.	Faster, because it deallocates the data pages used by the table, which is a minimal logging operation.
Transaction Log	Makes an entry in the transaction log for each deleted row.	Makes a single entry in the transaction log for the deallocation of pages.
Rollback	Since each deletion is logged, a ‘DELETE‘ operation can be easily rolled back using ‘ROLLBACK‘.	Cannot be rolled back in some database systems. In others, a rollback is possible but less straightforward than for DELETE.
Triggers	DML triggers (e.g., ‘ON DELETE‘) will be fired for each row that is deleted.	DDL triggers are not fired.
Identity Reset	Does not reset the value of an identity column. New rows will continue from the last identity value.	Resets the identity column of the table back to its seed value.

Quick Tip

Use ‘DELETE‘ when you need to remove specific rows or when you need the operation to be reversible. Use ‘TRUNCATE‘ when you want to quickly and completely empty a large table and reset its identity counter. Think of ‘DELETE‘ as erasing with a pencil (row by row) and ‘TRUNCATE‘ as wiping the entire board clean.

---

25. Write a program in C++ to generate and display prime number from 1 to 51.

Solution:

Step 1: Logic

The program will iterate through each number from 1 to 51. For each number, it will check if it is a prime number. A number is prime if it is greater than 1 and has no positive divisors other than 1 and itself. A helper function ‘isPrime‘ will be created for this check.

Step 2: C++ Program

```
#include <iostream>
#include <cmath>

// Function to check if a number is prime
bool isPrime(int n) {
    // 1 and numbers less than 1 are not prime
    if (n <= 1) {
        return false;
    }

    // Check for divisors from 2 up to the square root of n
    for (int i = 2; i <= sqrt(n); ++i) {
        if (n % i == 0) {
            // If a divisor is found, it's not a prime number
            return false;
        }
    }

    // If no divisors were found, it is a prime number
    return true;
}

int main() {
    std::cout << "Prime numbers from 1 to 51 are:" << std::endl;

    // Loop through numbers from 1 to 51
    for (int i = 1; i <= 51; ++i) {
        // Call the isPrime function to check the number
        if (isPrime(i)) {
            // If it's a prime, print it
            std::cout << i << " ";
        }
    }

    std::cout << std::endl;

    return 0;
}
```

Step 3: Expected Output

```
Prime numbers from 1 to 51 are:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47
```

Quick Tip

For efficiency in the ‘isPrime‘ function, it's only necessary to check for divisors up to the square root of the number. If a number ‘n‘ has a divisor larger than its square root, it must also have a corresponding divisor smaller than its square root.

---

26. Write a program in C++ to accept a number then display how many digits it contains and the sum of its digits.

Solution:

Step 1: Logic

The program will prompt the user to enter an integer. It will then use a ‘while‘ loop to process the number. In each iteration of the loop, it will:

1. Increment a counter for the number of digits.
2. Extract the last digit using the modulo operator (‘
3. Add this digit to a running sum.
4. Remove the last digit from the number using integer division (‘/ 10‘).

The loop continues until the number becomes 0. A special case for the input ‘0‘ must be handled.

Step 2: C++ Program

```
#include <iostream>

int main() {
    int number, originalNumber;
    int digitCount = 0;
    int sumOfDigits = 0;

    // Prompt user for input
    std::cout << "Enter an integer: ";
    std::cin >> number;

    // Store the original number for display later
    originalNumber = number;

    // Handle the special case of 0
    if (number == 0) {
        digitCount = 1;
    } else {
        // Handle negative numbers by using their absolute value
        if (number < 0) {
            number = -number;
        }

        // Loop until the number becomes 0
        while (number > 0) {
            // Increment the digit count
            digitCount++;

            // Get the last digit and add it to the sum
            sumOfDigits += number % 10;

            // Remove the last digit
            number /= 10;
        }
    }

    // Display the results
    std::cout << "The number " << originalNumber << " contains "
              << digitCount << " digits." << std::endl;
    std::cout << "The sum of its digits is " << sumOfDigits << "." << std::endl;

    return 0;
}
```

Step 3: Example Execution

```
Enter an integer: 472
```

```
The number 472 contains 3 digits.
The sum of its digits is 13.
```

Quick Tip

The combination of the modulo operator (‘