Multiple Choice Questions

1. NOT Gate is having how many input

A. One or more B. Two or more C. Zero or more D. One

2. How many combination are there for three input gates

A. 2 B. 4 C. 6 D. 8

3. Any Gate is having how many output

A. One or more B. Two or more C. One D. Zero or more

4. Which Law it is? X ∙ Y + X ∙ Z = ( X + Z ) ∙ ( X + Y )

A. Distributive Law B. Associative Law C. Transposition Law D. Commutative Law

5. Logically, the output of a NOR gate would have the same Boolean expression as a(n):

A. NAND gate immediately followed by an inverter

B. OR gate immediately followed by an inverter

C. AND gate immediately followed by an inverter

D. NOR gate immediately followed by an inverter

6. The Boolean expression for a 3-input AND gate is \_\_\_\_\_\_\_\_.

A. X = AB B. X = ABC C. X = A + B + C D. X = AB + C

7. What does the small bubble on the output of the NAND gate logic symbol mean?

A. open collector output

B. tristate

C. The output is inverted

D. none of the above

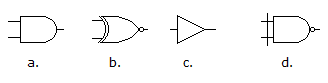
8. Which of the following gates has the exact inverse output of the OR gate for all possible input combinations?

A. NOR B. NOT C. NAND D. AND

9. Which of the following gates is described by the expression X= ABCD?

A. OR B. AND C. NOR D. NAND

10. Which of the figures shown below represents the exclusive-NOR gate?



A. a B. b C. c D. d

11. How many data select lines are required for selecting eight inputs?

A. 1 B. 2 C. 3 D. 4

12. Give the relationship that represents the dual of the Boolean property A + 1 = 1?

(Note: \* = AND, + = OR and ' = NOT)

A. A \* 1 = 1

B. A \* 0 = 0

C. A + 0 = 0

D. A \* A = A

E. A \* 1 = 1

13. Give the best definition of a literal?

A. A Boolean variable

B. The complement of a Boolean variable

C. 1 or 2

D. A Boolean variable interpreted literally

E. The actual understanding of a Boolean variable

14. Simplify the Boolean expression (A+B+C)(D+E)' + (A+B+C)(D+E) and choose the best answer.

A. A + B + C

B. D + E

C. A'B'C'

D. D'E'

E. None of the above

15. Which of the following relationships represents the dual of the Boolean property x + x'y = x + y?

A. x'(x + y') = x'y'

B. x(x'y) = xy

C. x\*x' + y = xy

D. x'(xy') = x'y'

E. x(x' + y) = xy

16. Given the function F(X,Y,Z) = XZ + Z(X'+ XY), the equivalent most simplified Boolean representation for F is:

A. Z + YZ B. Z + XYZ C. XZ

D. X + YZ E. None of the above

17. Which of the following Boolean functions is algebraically complete?

A. F = xy

B. F = x + y

C. F = x'

D. F = xy + yz

E. F = x + y'

18. Simplification of the Boolean expression (A + B)'(C + D + E)' + (A + B)' yields which of the following results?

A. A + B

B. A'B'

C. C + D + E

D. C'D'E'

E. A'B'C'D'E'

19. Given that F = A'B'+ C'+ D'+ E', which of the following represent the only correct expression for F'?

A. F'= A+B+C+D+E

B. F'= ABCDE

C. F'= AB(C+D+E)

D. F'= AB+C'+D'+E'

E. F'= (A+B)CDE

20. An equivalent representation for the Boolean expression A' + 1 is

A. A B. A' C. 1 D. 0

21. Simplification of the Boolean expression AB + ABC + ABCD + ABCDE + ABCDEF yields which of the following results?

A. ABCDEF

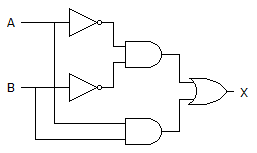
B. AB

C. AB + CD + EF

D. A + B + C + D + E + F

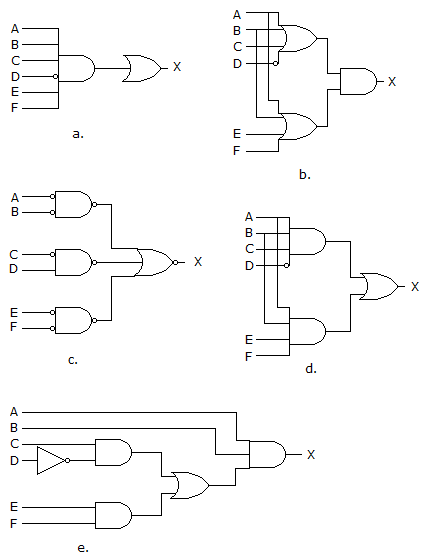
E. A + B(C+D(E+F))

22. Which of the following Boolean expressions shows the diagram below?



|  |  |
| --- | --- |
| [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mca4_1018a1.jpeg |
| [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mca4_1018b1.jpeg |
| [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mca4_1018c1.jpeg |
| [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mca4_1018d1.jpeg |

23. Which of the circuits in figure (a to d) is the sum-of-products implementation of figure (e)?



|  |  |
| --- | --- |
| 24. | The implementation of simplified sum-of-products expressions may be easily implemented into actual logic circuits using all universal \_\_\_\_\_\_\_\_ gates with little or no increase in circuit complexity. (Select the response for the blank space that will BEST make the statement true.) |
|  | |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | AND/OR | | [**B.**](javascript:%20void%200;) | NAND | | [**C.**](javascript:%20void%200;) | NOR | | [**D.**](javascript:%20void%200;) | OR/AND | |
|  | 25. Which of the figures in figure (a to d) is equivalent to figure (e)?  https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mcq4_1017_1.gifhttps://www.indiabix.com/_files/images/digital-electronics/digital-systems/mcq4_1017_1.gif |
| |  |  | | --- | --- | | [A.](javascript:%20void%200;) | a | | [B.](javascript:%20void%200;) | b | | [C.](javascript:%20void%200;) | c | | [D.](javascript:%20void%200;) | d | |

|  |  |
| --- | --- |
| 26. | How many data select lines are required for selecting eight inputs? |
| |  |  | | --- | --- | | [A.](javascript:%20void%200;) | 1 | | [B.](javascript:%20void%200;) | 2 | | [C.](javascript:%20void%200;) | 3 | | [D.](javascript:%20void%200;) | 4 | |

|  |  |
| --- | --- |
| 27. | Which of the following logic expressions represents the logic diagram shown?  https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mcq4_1018_1.gif |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mca4_1018a1.jpeg | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mca4_1018b1.jpeg | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mca4_1018c1.jpeg | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mca4_1018d1.jpeg | |

|  |  |
| --- | --- |
| 28. | The implementation of simplified sum-of-products expressions may be easily implemented into actual logic circuits using all universal \_\_\_\_\_\_\_\_ gates with little or no increase in circuit complexity. (Select the response for the blank space that will BEST make the statement true.) |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | AND/OR | | [**B.**](javascript:%20void%200;) | NAND | | [**C.**](javascript:%20void%200;) | NOR | | [**D.**](javascript:%20void%200;) | OR/AND | |
| 29. | Which gate is best used as a basic comparator? |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | NOR | | [**B.**](javascript:%20void%200;) | OR | | [**C.**](javascript:%20void%200;) | Exclusive-OR | | [**D.**](javascript:%20void%200;) | AND | |
|  | 30. Which of the following expressions is in the product-of-sums form? |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | (*A* + *B*)(*C* + *D*) | | [**B.**](javascript:%20void%200;) | (*AB*)(*CD*) | | [**C.**](javascript:%20void%200;) | *AB*(*CD*) | | [**D.**](javascript:%20void%200;) | *AB* + *CD* | |

|  |  |
| --- | --- |
| 31. | Which of the following is an important feature of the sum-of-products form of expressions? |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | All logic circuits are reduced to nothing more than simple AND and OR operations. | | [**B.**](javascript:%20void%200;) | The delay times are greatly reduced over other forms. | | [**C.**](javascript:%20void%200;) | No signal must pass through more than two gates, not including inverters. | | [**D.**](javascript:%20void%200;) | The maximum number of gates that any signal must pass through is reduced by a factor of two. | |

|  |  |
| --- | --- |
| 32. | Which of the following expressions is in the sum-of-products form? |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | (*A* + *B*)(*C* + *D*) | | [**B.**](javascript:%20void%200;) | (*AB*)(*CD*) | | [**C.**](javascript:%20void%200;) | *AB*(*CD*) | | [**D.**](javascript:%20void%200;) | *AB* + *CD* | |

|  |  |
| --- | --- |
| 33. | Solve the network in the figure given below for *X*.  https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mcq4_1009_1.gif |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | *A* + *BC* + *D* | | [**B.**](javascript:%20void%200;) | ((*A* + *B*)*C*) + *D* | | [**C.**](javascript:%20void%200;) | *D*(*A* + *B* + *C*) | | [**D.**](javascript:%20void%200;) | (*AC* + *BC*)*D* | |

|  |  |
| --- | --- |
| 34. | What type of logic circuit is represented by the figure shown below?  https://www.indiabix.com/_files/images/digital-electronics/digital-systems/mcq4_1018_1.gif |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | XOR | | [**B.**](javascript:%20void%200;) | XNOR | | [**C.**](javascript:%20void%200;) | XAND | | [**D.**](javascript:%20void%200;) | XNAND | |

|  |  |
| --- | --- |
| 1. | Convert the following SOP expression to an equivalent POS expression.  https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_01600.gif |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0160a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0160b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0160c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0160d.gif | |
| 2. | Determine the values of A, B, C, and D that make the sum term https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_00100.gif equal to zero. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | A = 1, B = 0, C = 0, D = 0 | | [**B.**](javascript:%20void%200;) | A = 1, B = 0, C = 1, D = 0 | | [**C.**](javascript:%20void%200;) | A = 0, B = 1, C = 0, D = 0 | | [**D.**](javascript:%20void%200;) | A = 1, B = 0, C = 1, D = 1 | |
| 3. | Which of the following expressions is in the sum-of-products (SOP) form? |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | (*A* + *B*)(*C* + *D*) | | [**B.**](javascript:%20void%200;) | (*A*)*B*(*CD*) | | [**C.**](javascript:%20void%200;) | *AB*(*CD*) | | [**D.**](javascript:%20void%200;) | *AB* + *CD* | |
| 4. | Derive the Boolean expression for the logic circuit shown below:  https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_00900.gif |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0090a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0090b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0090c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0090d.gif | |
| 5. | From the truth table below, determine the standard SOP expression.  https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_02100.gif |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0210a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0210b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0210c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0210d.gif | |

Top of Form

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 6. | One of De Morgan's theorems states that https://www.indiabix.com/_files/images/digital-electronics/basic-digital-electronics/mcq5_1017_1u.jpg. Simply stated, this means that logically there is no difference between: | | | | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | a NOR and an AND gate with inverted inputs | | [**B.**](javascript:%20void%200;) | a NAND and an OR gate with inverted inputs | | [**C.**](javascript:%20void%200;) | an AND and a NOR gate with inverted inputs | | [**D.**](javascript:%20void%200;) | a NOR and a NAND gate with inverted inputs | | | | | | | |
| 7. | The commutative law of Boolean addition states that *A* + *B* = *A* × *B*. | | | | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | | | | | | | |
| 8. | | Applying DeMorgan's theorem to the expression https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_00400.gif, we get \_\_\_\_\_\_\_\_. | | | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0040a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0040b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0040c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0040d.gif | | | | | | |
| 9. | | | The systematic reduction of logic circuits is accomplished by: | | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | using Boolean algebra | | [**B.**](javascript:%20void%200;) | symbolic reduction | | [**C.**](javascript:%20void%200;) | TTL logic | | [**D.**](javascript:%20void%200;) | using a truth table | | | | | |
| 10. | | | | Which output expression might indicate a product-of-sums circuit construction? | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/basic-digital-electronics/mca5_1023a1.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/basic-digital-electronics/mca5_1023b1.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/basic-digital-electronics/mca5_1023c1.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/basic-digital-electronics/mca5_1023d1.gif | | | | |
| 11. | | | | An AND gate with schematic "bubbles" on its inputs performs the same function as a(n)\_\_\_\_\_\_\_\_ gate. | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | NOT | | [**B.**](javascript:%20void%200;) | OR | | [**C.**](javascript:%20void%200;) | NOR | | [**D.**](javascript:%20void%200;) | NAND | | | | |
| 12. | | | | For the SOP expression https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_01800.gif, how many 1s are in the truth table's output column? | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | 1 | | [**B.**](javascript:%20void%200;) | 2 | | [**C.**](javascript:%20void%200;) | 3 | | [**D.**](javascript:%20void%200;) | 5 | | | | |
| 13. | | | | A truth table for the SOP expression https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_01700.gif has how many input combinations? | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | 1 | | [**B.**](javascript:%20void%200;) | 2 | | [**C.**](javascript:%20void%200;) | 4 | | [**D.**](javascript:%20void%200;) | 8 | | | | |
| 14. | | | | How many gates would be required to implement the following Boolean expression before simplification? XY + X(X + Z) + Y(X + Z) | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | 1 | | [**B.**](javascript:%20void%200;) | 2 | | [**C.**](javascript:%20void%200;) | 4 | | [**D.**](javascript:%20void%200;) | 5 | | | | |
| 15. | | | | Determine the values of A, B, C, and D that make the product term https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_00200.gif equal to 1. | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | A = 0, B = 1, C = 0, D = 1 | | [**B.**](javascript:%20void%200;) | A = 0, B = 0, C = 0, D = 1 | | [**C.**](javascript:%20void%200;) | A = 1, B = 1, C = 1, D = 1 | | [**D.**](javascript:%20void%200;) | A = 0, B = 0, C = 1, D = 0 | | | | |
| 16. | | | | What is the primary motivation for using Boolean algebra to simplify logic expressions? | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | It may make it easier to understand the overall function of the circuit. | | [**B.**](javascript:%20void%200;) | It may reduce the number of gates. | | [**C.**](javascript:%20void%200;) | It may reduce the number of inputs required. | | [**D.**](javascript:%20void%200;) | all of the above | | | | |
| 17. | | | | How many gates would be required to implement the following Boolean expression after simplification? XY + X(X + Z) + Y(X + Z) | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | 1 | | [**B.**](javascript:%20void%200;) | 2 | | [**C.**](javascript:%20void%200;) | 4 | | [**D.**](javascript:%20void%200;) | 5 | | | | |
| 18. | | | | *AC* + *ABC* = *AC* | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | | | | |
| 19. | | | | When https://www.indiabix.com/_files/images/digital-electronics/basic-digital-electronics/mcq5_1014_1.jpg are the inputs to a NAND gate, according to De Morgan's theorem, the output expression could be: | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | *X* = *A* + *B* | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/basic-digital-electronics/mca5_1014b1.gif | | [**C.**](javascript:%20void%200;) | *X* = (*A*)(*B*) | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/basic-digital-electronics/mca5_1014d1.gif | | | | |
| 20. | | | | Which Boolean algebra property allows us to group operands in an expression in any order without affecting the results of the operation [for example, *A* + *B* = *B* + *A*]? | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | associative | | [**B.**](javascript:%20void%200;) | commutative | | [**C.**](javascript:%20void%200;) | Boolean | | [**D.**](javascript:%20void%200;) | distributive | | | | |
| 21. | | | | Applying DeMorgan's theorem to the expression https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mcq4_00700.gif, we get \_\_\_\_\_\_\_\_ | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0070a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0070b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0070c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/mca4_0070d.gif | | | | |
|  | | | | | | |
|  | | | | | | |
| 23. | | | | Use Boolean algebra to find the most simplified SOP expression for *F* = *ABD* + *CD* + *ACD* + *ABC* + *ABCD*. | | | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | *F* = *ABD* + *ABC* + *CD* | | [**B.**](javascript:%20void%200;) | *F* = *CD* + *AD* | | [**C.**](javascript:%20void%200;) | *F* = *BC* + *AB* | | [**D.**](javascript:%20void%200;) | *F* = *AC* + *AD* | | | | |
|  | | | | |  | | |
|  | | |
| 25. | | | | | | The NAND or NOR gates are referred to as "universal" gates because either: | |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | can be found in almost all digital circuits | | [**B.**](javascript:%20void%200;) | can be used to build all the other types of gates | | [**C.**](javascript:%20void%200;) | are used in all countries of the world | | [**D.**](javascript:%20void%200;) | were the first gates to be integrated | | |

Exercise :: Boolean Algebra and Logic Simplification - True or False

|  |  |
| --- | --- |
| 1. | A variable is a symbol used to represent a logical quantity that can have a value of 1 or 0. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 2. | The OR function is Boolean multiplication and the AND function is Boolean addition. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 3. | In Boolean algebra, A + 1 = 1. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 4. | The product-of-sums (POS) is basically the ORing of ANDed terms. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 5. | In Boolean algebra, https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/aoverbar2.gif = A. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 6. | SOP stands for sum-of-powers. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 7. | The VHDL editor provided with a schematic editor development system will produce a file with the extension .vhd, which can be used by the simulator to test the output of the logic design. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 8. | In the commutative law, in ORing and ANDing of two variables, the order in which the variables are ORed or ANDed makes no difference. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 9. | The binary value of 1010 is converted to the product term https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/tfq4_00800.gif. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 10. | Most Boolean reductions result in an equation in only one form. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 11. | The symbol shown below is for a 2-input NAND gate.  https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/tfq4_00300.gif |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 12. | A Karnaugh map is similar to a truth table because it presents all the possible values of input variables and the resulting output of each value. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 13. | The process of reduction or simplification of combinational logic circuits increases the cost of the circuit. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 14. | By applying De Morgan's theorem to a NOR gate, two identical truth tables can be produced. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |
| 15. | Five-variable Karnaugh maps are impossible. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | True | | [**B.**](javascript:%20void%200;) | False | |

Exercise :: Boolean Algebra and Logic Simplification - Filling the Blanks

|  |  |
| --- | --- |
| 1. | The Boolean expression C + CD is equal to \_\_\_\_\_\_\_\_. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | C | | [**B.**](javascript:%20void%200;) | D | | [**C.**](javascript:%20void%200;) | C + D | | [**D.**](javascript:%20void%200;) | 1 | |
| 2. | The Boolean expression for the logic circuit shown is \_\_???\_\_\_.  https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cuq4_00500.gif |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0050a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0050b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0050c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0050d.gif | |
| 3. | Applying DeMorgan's theorem and Boolean algebra to the expression https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cuq4_00400u.png results in \_\_\_\_\_\_\_\_. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0040a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0040bu.png | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0040c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0040d.gif | |
| 4. | The standard SOP form of the expression https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cuq4_00600.gif is \_\_\_\_\_\_\_\_. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0060a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0060b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0060c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0060d.gif | |

|  |  |
| --- | --- |
| 5. | Identify the Boolean expression that is in standard POS form. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0080a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0080b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0080c.gif | | [**D.**](javascript:%20void%200;) | (A + B)(C + D) | |
| 6. | The Boolean expression https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cuq4_00300.gif is equal to \_\_\_\_\_\_\_\_. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | C | | [**B.**](javascript:%20void%200;) | D | | [**C.**](javascript:%20void%200;) | C + D | | [**D.**](javascript:%20void%200;) | 1 | |
| 7. | The Boolean expression https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cuq4_00700.gif can be reduced to \_\_\_\_\_\_\_\_. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0070a.gif | | [**B.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0070b.gif | | [**C.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0070c.gif | | [**D.**](javascript:%20void%200;) | https://www.indiabix.com/_files/images/digital-electronics/digital-fundamentals/cua4_0070d.gif | |
| 8. | Which of the following is true for a 5-variable Karnaugh map? |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | There is no such thing. | | [**B.**](javascript:%20void%200;) | It can be used only with the aid of a computer. | | [**C.**](javascript:%20void%200;) | It is made up of two 4-variable Karnaugh maps. | | [**D.**](javascript:%20void%200;) | It is made up of a 2-variable and a 3-variable Karnaugh map. | |
| 9. | When four 1s are taken as a group on a Karnaugh map, the number of variables eliminated from the output expression is \_\_\_\_\_\_\_\_. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | 1 | | [**B.**](javascript:%20void%200;) | 2 | | [**C.**](javascript:%20void%200;) | 3 | | [**D.**](javascript:%20void%200;) | 4 | |

|  |  |
| --- | --- |
| 10. | In Boolean algebra, the word "literal" means \_\_\_\_\_\_\_\_. |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | a product term | | [**B.**](javascript:%20void%200;) | all the variables in a Boolean expression | | [**C.**](javascript:%20void%200;) | the inverse function | | **[D.](javascript:%20void%200;)** | a variable or its complement | |