Boolean Logic

Boolean Logic is a concept that involves binary variables and operations. It focuses on the values true and false (1 and 0). It was developed by the English Mathematician and logician George Boole. Boolean Algebra comprises of following:

1. Boolean Expression
2. Boolean Variable

Boolean Variable

A boolean variable is a variable that holds boolean values True/ False or 1/0.

Boolean Expression

A boolean expression is an expression that consists of a combination of boolean variables, boolean values and boolean operators. A boolean expression evaluates to either True or False.

Boolean Operators

Boolean operators perform operations on operands (Boolean Variables/Values). The boolean operators are: AND, OR and NOT.

AND operator - It evaluates to True(1) if all inputs are True(1), otherwise False(0). It is represented by the dot operator (.)
Example : A.B may be read A AND B

OR operator - It evaluates to True(1) if any of the inputs is True(1), otherwise False(0). It is represented by the + operator.
Example: A + B may be read as A OR B.

NOT operator - It evaluates to True(1) when the condition is False, and returns False(0) when the condition is True(1).

Example: A' may be read as A complement.
$\bar{A}$ may be read as A complement.

Truth Table

A truth table is a representation of all possible combinations of the input variables and the corresponding output values.

The number of rows in a truth table are 2^n, where n is no. of input variables.

Example- Here, A and B are the input boolean variables, OR (+) is the operator, F is the output.

A	B	F = A OR B
0	0	0
0	1	1
1	0	1
1	1	1

Rules of Boolean Logic (Additional Reading)

Name of Rule	**AND Version**	**OR Version**
Identity Law	1.A=A	0+A=A
Null Law or Dominant Law	0.A=0	1+A=1
Idempotent Law	A.A=A	A+A=A
Inverse Law	A.A’=0	A+A’=1
Commutative Law	AB=BA	A+B=B+A
Associative Law	(AB)C=A(BC)	(A+B)+C=A+(B+C)
Distributive Law	A+BC=(A+B)(A+C)	A(B+C)=AB+AC
Absorption Law	A(A+B)=A	A+AB=A
De Morgan’s Law	(AB)’=A’+B’	(A+B)’=A’.B’

De Morgan’s Laws

DeMorgan’s First Law states that when two input variables are AND’ed and complemented, they are equivalent to the OR of the complements of the individual variables.

(AB)'=A'+B'

A	B	A'	B'	AB	**(AB)'**	**A'+B'**
0	0	1	1	0	**1**	**1**
0	1	1	0	0	**1**	**1**
1	0	0	1	0	**1**	**1**
1	1	0	0	1	**0**	**0**

DeMorgan's Second Law states that when two input variables are OR'ed and complemented, they are equivalent to the AND of the complements of the individual variables.

(A+B)'=A'.B'

A	B	A'	B'	A+B	**(A+B)'**	**A'.B'**
0	0	1	1	0	**1**	**1**
0	1	1	0	1	**0**	**0**
1	0	0	1	1	**0**	**0**
1	1	0	0	1	**0**	**0**

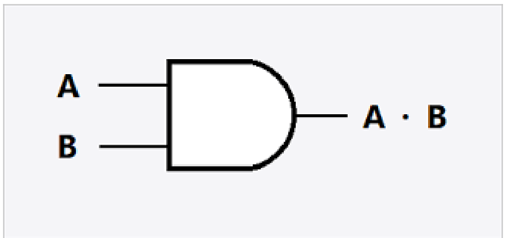
Logic Gates

A logic gate is a device that performs a Boolean Function. One or more inputs in the form of 1/0 are provided to get the specific output governed by a logic. Examples of Logic Gates are: AND, OR, NOT, NOR, NAND and XOR.

NAND and NOR are called Universal Gates, as they can implement any Boolean expression.

AND Gate

Logic Gate Diagram

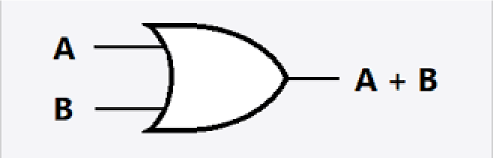


Truth Table

A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

OR Gate

Logic Gate Diagram

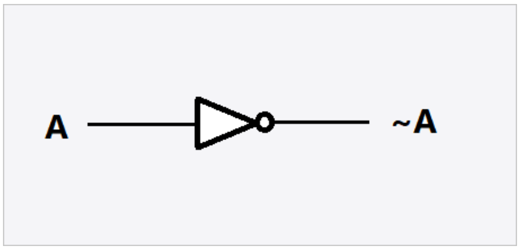


Truth Table

A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

NOT Gate

Logic Gate Diagram

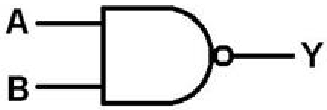


Truth Table

A	A'
0	1
1	0

NAND Gate

Logic Gate Diagram

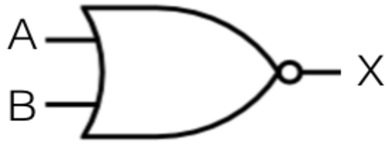


Truth Table

A	B	A.B	(A.B)'
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

NOR Gate

Logic Gate Diagram

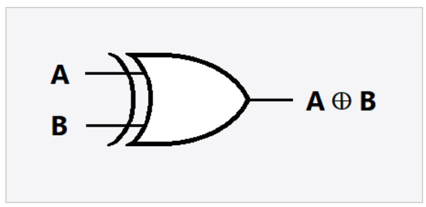


A	B	A+B	(A+B)'
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Truth Table

XOR Gate

Logic Gate Diagram



Truth Table

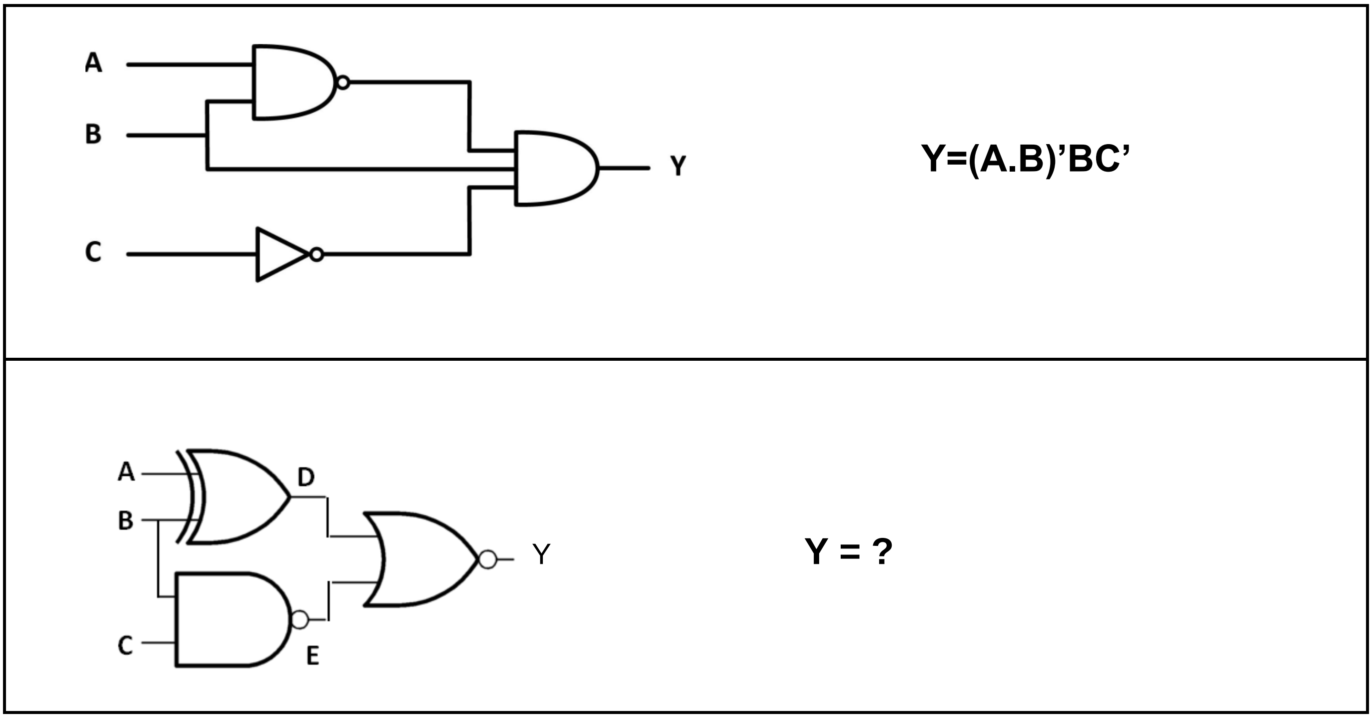
A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

XOR is represented as AB' + A'B

Logic Circuits

A logic circuit is an electronic circuit which performs logical operations on the input boolean variables, and transforms them into the output using a combination of Logic Gates.

Examples of a circuit diagram are:



Number System

There are many possible ways to represent numbers. A number system provides a consistent and unique method to represent the numbers.

Positional Number System

A positional number system is one way of writing numbers. It has unique symbols for 0 through (b – 1), where b is the base (also known as radix) of the system. These symbols are called digits.

Some popular positional number systems are:

1. Binary - base 2
2. Octal - base 8
3. Decimal - base 10
4. Hexadecimal - base 16

Decimal Number System

A decimal number uses base-10 for representing numbers. The numbers 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9 are the digits present in the system. Every number can be represented as a sum of powers of 10, with each power weighted by a digit.

For example, in the decimal system, 534 can be represented as

$$534 = 5 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

Fractional numbers can also be represented using similar notation, for example, 123.238 can be represented as

$$123.238 = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 2 \times 10^{-1} + 3 \times 10^{-2} + 8 \times 10^{-3}$$

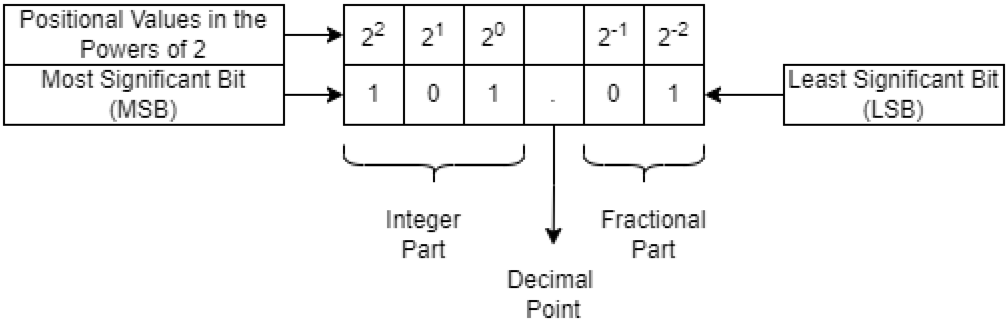
The powers of 10, which correspond respectively to the digits in a decimal integer when read from right to left, are called the place values of the digits.

Binary Number System

The binary number system uses only two symbols, 0 and 1. Each numeral is known as a binary digit or a bit. It is also known as the base-2 numeral system, where the positional digits are powers of 2.

The binary system is most easily implemented through computer hardware. The binary digits 0 and 1 correspond to the presence or the absence of a digital signal. The manipulation of binary digits is performed using a combination of boolean logic gates and circuits. Negative numbers can also be represented in the Binary Number System.

The numbers are represented by creating a sequence of bits, such as 00000, 010100, 011100 etc. Binary numerals are often subscripted with 2 in order to indicate the base. For example, $(110101)_2$. Numbers that have no fractional part are called binary integers.



In the figure above, representation of the binary number $(101.01)_2$ is shown. The most significant bit (MSB) is the bit in a binary number sequence with the largest value. This is usually the bit farthest to the left. The MSB represents the highest-order place of the binary integer. On the other hand, the Least Significant Bit (LSB) is the right-most bit of the number.

The maximum possible sequence of binary numbers that can be created using a sequence of n-bits is 2^n .

No. of Bits	Count of possible binary numbers
1	$2^1 = 2$
2	$2^2 = 4$
3	$2^3 = 8$
4	$2^4 = 16$

Octal Number System

Octal system, or the base-8 system uses 8 unique symbols, 0, 1, 2, 3, 4, 5, 6, and 7. The place values in the octal system are powers of 8. We can use 3-bit binary numbers to represent the octal digits, since $2^3 = 8$.

Decimal Number	3-bit Binary Number	Octal Number
0	000	0
1	001	1
2	010	2
3	011	3
4	100	4
5	101	5
6	110	6
7	111	7

For example, an octal number $(532.67)_8$ can be represented in the decimal number system as

$$(532.67)_8 = 5 \times 8^2 + 3 \times 8^1 + 2 \times 8^0 + 6 \times 8^{-1} + 7 \times 8^{-2}$$

Octal numbers find applications where the number of bits in one word is a multiple of 3. They are also used as shorthand for representing file permissions on UNIX/Linux operating systems and representation of UTF8 numbers, etc. The advantage of using Octal numbers is that it uses fewer digits than the decimal number system. It has fewer computations and is less prone to the computational errors.

Hexadecimal Number System

The hexadecimal number system uses 16 unique symbols, 0-9 and A-F for representation of numbers. The place values in the hexadecimal system are powers of 16. Each hexadecimal digit has a unique 4-bit representation since $2^4 = 16$, so all possible digits can be represented with 4 bit binary numbers. The hexadecimal numbering system is often used by programmers to simplify the binary numbering system since large binary numbers can be organized and written neatly in their respective hexadecimal notations.

Decimal Number	4-Bit Binary Number	Hexadecimal Number
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Some popular uses of Hexadecimal numbers are:

- Representation of addresses in memory.
- Representation of colors on web pages.
- Representation of Media Access Control (MAC) addresses.
- Display error messages.

Number System Conversions

1) **Decimal Number System to Other Base**
   a) Decimal Number System to Binary Number System
   b) Decimal Number System to Octal Number System
   c) Decimal Number System to Hexadecimal Number System
2) **Other Base to Decimal Number System**
   a) Binary Number System to Decimal Number System
   b) Octal Number System to Decimal Number System
   c) Hexadecimal Number System to Decimal Number System
3) **One Base to Other Base System**
   a) Binary Number System to Hexadecimal Number System
   b) Hexadecimal Number System to Binary Number System
   c) Binary Number System to Octal Number System
   d) Octal Number System to Binary Number System
   e) Hexadecimal Number System to Octal Number System
   f) Octal Number System to Hexadecimal Number System

Decimal Number System to Binary Number System Conversion

Steps for the Integer part

1. For the integral part of the number, repeatedly divide the number by 2, and save the remainder.
2. Repeat the process until no further division is possible.
3. Read the remainders from the bottom to top to get the binary number.

Steps for the fractional part

1. Repeatedly multiply the fractional part by 2 and note the integer and fractional part of the product.
2. Repeatedly multiply the fraction part by 2 and note the integer part from top to the bottom. (Stop if you get 0)
3. Repeat this procedure till sufficient precision is reached (usually 2 to 3 places after the radix point).

Ques 1 - Convert decimal number 75 to binary number.

Divisor	Dividend	Remainder	
2	75		
2	37	1	LSB
2	18	1	
2	9	0	
2	4	1	
2	2	0	
2	1	0	
	0	1	MSB

$=(1001011)_2$

Ques 2- Convert decimal number 142.67 to a binary number.

Divisor	Dividend	Remainder	
2	142		
2	71	0	LSB
2	35	1	
2	17	1	
2	8	1	
2	4	0	
2	2	0	
2	1	0	
	0	1	MSB

.67 x **2 = 1.34**	**1**	**.34**
.34 x **2 = 0.68**	**0**	**.68**
.68 x **2 = 1.36**	**1**	**.36**
.36 x **2 = 0.72**	**0**	**.72**

= **$(10001110.1010)_2$**

Decimal Number System to Octal Number System Conversion

Steps for the Integer part

1. For the integral part of the number, repeatedly divide the number by 8, and save the remainder.
2. Repeat the process until no further division is possible.
3. Read the remainders from the bottom to top to get the binary number.

Steps for the fractional part

1. Repeatedly multiply the fractional part by 8 and note the integer and fractional part of the product.
2. Repeatedly multiply the fraction part by 8 and note the integer part from top to the bottom. (Stop if you get 0)
3. Repeat this procedure till sufficient precision is reached (usually 2 to 3 places after the radix point).

Ques 1 - Convert decimal number 104 to octal number.

Divisor	Dividend	Remainder	
8	104		
8	13	0	LSB
8	1	5	
	0	1	MSB

= $(150)_8$

Ques 2 - Convert decimal number 563.93 to octal number.

Divisor	Dividend	Remainder	
8	563		
8	70	3	LSB
8	8	6	
8	1	0	
	0	1	MSB

.93 x **8 = 7.44**	**7**	**.44**
.44 x **8 = 0.88**	**0**	**.88**
.88 x **8 = 7.04**	**7**	**.04**
.04 x **8 = 0.32**	**0**	**.32**

= $(1063.7070)_8$

Decimal Number System to Hexadecimal Number System Conversion

Steps for the Integer part

1. For the integral part of the number, repeatedly divide the number by 16, and save the remainder.
2. Repeat the process until no further division is possible.
3. Read the remainders from the bottom to top to get the binary number.

Steps for the fractional part

1. Repeatedly multiply the fractional part by 16 and note the integer and fractional part of the product.
2. Repeatedly multiply the fraction part by 16 and note the integer part. (Stop if you get 0)

3. Repeat this procedure till sufficient precision is reached (usually 2 to 3 places after the radix point).

Ques 1 - Convert decimal number 600 to hexadecimal number.

Divisor	Dividend	Remainder	
16	600		
16	37	8	LSB
16	2	5	
	0	2	MSB

$= (258)_{16}$

Ques 2 - Convert decimal number 49 to hexadecimal number.

Divisor	Dividend	Remainder	
16	49		
16	3	1	LSB
	0	3	MSB

$= (31)_{16}$

Binary Number System to Decimal Number System Conversion

Steps:

1. Multiply each digit with its respective weighted power of 2.
2. Calculate its sum.

Ques 1 - Convert Binary Number 101101 into Decimal Number.

101101

$= 1x2^5 + 0x2^4 + 1x2^3 + 1x2^2 + 0x2^1 + 1x2^0$

= 32 + 0 + 8 + 4 + 0 + 1

= 45

Ques 2 - Convert Binary Number 110101.010 into Decimal Number.

110101.010

$= 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3}$

= 32 + 16 + 0 + 4 + 0 + 1 + 0 + 0.25 + 0

= 53.25

Octal Number System to Decimal Number System Conversion

Steps:

1. Multiply each digit with its respective weighted power of 8.
2. Calculate its sum.

Ques 1 - Convert Octal Number 156 into Decimal Number.

156

$= 1 \times 8^2 + 5 \times 8^1 + 6 \times 8^0$

$= 64 + 40 + 6$

$= 110$

Ques 2 - Convert Octal Number 78.12172 into Decimal Number.

78.12

$= 7 \times 8^1 + 8 \times 8^0 + 1 \times 8^{-1} + 2 \times 8^{-2}$

$= 56 + 8 + 0.125 + 0.03125$

$= 64.15625$

Hexadecimal Number System to Decimal Number System Conversion

Steps:

1. Multiply each digit with its respective weighted power of 16.
2. Calculate its sum.

Ques 1 - Convert Hexadecimal Number 0.87 into Decimal Number.

0.87

$= 0 \times 16^0 + 8 \times 16^{-1} + 7 \times 16^{-2}$

$= 0 + 0.5 + 0.02734375$

$= 0.52734375$

Binary Number System to Hexadecimal Number System

There are two ways to convert a binary number to a hexadecimal number:

1. Convert binary number to decimal number, and then convert to hexadecimal number.
2. Using grouping of bits.

Binary Number	Hexadecimal Number	Binary Number	Hexadecimal Number
0000	0	1000	8
0001	1	1001	9
0010	2	1010	A(10)
0011	3	1011	B(11)
0100	4	1100	C(12)
0101	5	1101	D(13)
0110	6	1110	E(14)
0111	7	1111	F(15)

For the integer part, make groups of four bits together starting from the left side of the radix point, and for the fractional part, make groups of four bits towards the right side of the radix point.
Replace each group with the corresponding hexadecimal number, we get the hexadecimal equivalent of the given binary number.

NOTE: Add any number of 0's in the left most bit for integer part, and 0's in right most bit for fractional part required to complete the grouping of 4 bits.

Ques 1 - Convert binary number 1000110101110 to hexadecimal number.
Converting Binary Number into Decimal Number:

$= (1000110101110)_2$
$= 1 \times 2^{12} + 0 \times 2^{11} + 0 \times 2^{10} + 0 \times 2^9 + 1 \times 2^8 + 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$
$= 4096 + 0 + 0 + 0 + 256 + 128 + 0 + 32 + 0 + 8 + 4 + 2 + 0$
$= (4526)_{10}$

Converting Decimal into Hexadecimal Number:

Divisor	Dividend	Remainder	
16	4526		
16	282	14(E)	LSB
16	17	10(A)	
16	1	1	
	0	1	MSB

$= (11AE)_{16}$

Ques 2 - Convert Binary number 10010111101101 into Hexadecimal Number.

0010	0101	1110	1101

2 5 E D

= $(25ED)_{16}$

Ques 3 - Convert Binary number 101100100.100111 into Hexadecimal Number.

0001	0110	0100	.	1001	1100
1	**6**	**4**		**9**	**C**

Hence the equivalent Hexadecimal Number is $(164.9C)_{16}$

Hexadecimal Number System to Binary Number System

A hexadecimal number can be converted to Binary Number through any of the two methods:

1. Converting Hexadecimal Number to Decimal Number, and then converting Decimal Number to Binary number.
2. Using Conversion Table

Ques 1 - Convert Hexadecimal Number 7B316 to Binary Equivalent Number.

7	B	3	1	6

0111 1011 0011 0001 0110

= 0111 1011 0011 0001 0110

Hence the equivalent binary number is $(0111101100110001 0110)_2$

Ques 2 - Convert Hexadecimal Number D2.92AB to Binary equivalent Number.

D	2	.	9	2	A	B

1101 0010 1001 0010 1010 1011

= 1101 0010 1001 0010 1010 1011

Hence the equivalent binary number is $(11010010.1001001010101011)_2$

Binary Number System to Octal Number System

There are two ways to convert a binary number to a hexadecimal number:

1. Convert binary number to decimal number or octal number, and then convert to octal number.
2. Using grouping of bits

Binary Number	Octal Number
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

For the integer part, make groups of three bits together starting from the left side of the radix point, and for the fractional part, make groups of three bits towards the right side of the radix point.

NOTE: Add any number of 0's in the left most bit for integer part, and 0's in the right most bit for fractional part required to complete the grouping of 3 bits.

Ques 1 - Convert binary number 1011011010 to octal number.

001	011	011	010
1	**3**	**3**	**2**

= $(1332)_8$

Octal Number System to Binary Number System

An Octal number can be converted to Binary Number through any of the two methods:

1. Converting Octal Number to Decimal Number, and then converting Decimal Number to Binary number.
2. Using Conversion Table

Ques 1 - Convert Octal Number 767 to Binary Equivalent Number.

7	6	7

111 110 111

= 111 110 111

Hence the equivalent binary number is $(111110111)_2$

Ques 2 - Convert Octal Number 4716.15 to Binary Equivalent Number.

4	7	1	6	.	1	5

100 111 001 110 001 101

= $(100111001110.001101)_2$

Hexadecimal Number System to Octal Number System

Steps:

1. Convert each digit of the hexadecimal number into a 4 bit binary equivalent number.
2. Combine 3 bits from the LSB, and convert each group of 3 bits into octal using a table.

Ques 1 - Convert Hexadecimal Number 4B2A into Octal Number.

4	B (11)	2	A (10)

0100 **1011** **0010** **1010**

= 0100 1011 0010 1010

= 0 100 101 100 101 010

= 000 100 101 100 101 010

= $(45452)_8$

Ques 2 - Convert Hexadecimal Number FE6 into Octal Number.

F	E	6

1111	1110	0110

= 1111 1110 0110

= 111 111 100 110

= $(7746)_8$

Octal Number System to Hexadecimal Number System

Steps:

1. Convert each digit of the octal number into a 3 bit binary equivalent number.
2. Combine 4 bits from the LSB, and convert each group of 4 bits into hexadecimal using a table.

Ques 1 - Convert Octal Number 2106 into Hexadecimal Number.

2	1	0	6

010 001 000 110

= 010 001 000 110

= 0100 0100 0110

= $(446)_{16}$

Ques 2 - Convert Octal Number 765 into Hexadecimal Number.

7	6	5

111 110 101

= 111 110 101

= 1 1111 0101

= 0001 1111 0101

= $(1F5)_{16}$

Encoding Schemes

Encoding is defined as the process to convert data from one form to another. Computers only understand binary language. There is a need to convert popularly used languages by humans into machine understandable format. Textual characters (letters, numbers and symbols) are assigned unique numerical codes. Some of the popular encoding schemes are ASCII, ISCII and Unicode.

ASCII

- ASCII stands for American Standard Code for Information Interchange.
- It was developed in the United States of America by American Standards Association (ASA).
- It is used to represent textual information in computers.
- ASCII uses 7-bits for encoding.
- A maximum of 128 characters may be encoded. Out of these 128 characters, only 95 are printable including the digits 0-9, lowercase and uppercase characters a-z A-Z and punctuation marks. The rest of the characters are Control Characters.

ISCII

- ISCII stands for Indian Script Code for Information Interchange (ISCII).
- ISCII is an extended version of the ASCII code and uses 8 bits for encoding.
- It was developed by the Bureau of Indian Standards, India.
- It represents various languages from India.
- Some of the supported scripts are Devanagari (Hindi, Marathi, Sanskrit, Konkani), Gujarati, Kannada, Punjabi, Malayalam, Telugu, Tamil, Oriya, Bengali-Assamese etc.

Unicode

- Unicode Standard is developed and managed by the Unicode Consortium, which is a non-profit organization composed of members from Software Development companies such as Apple, Adobe, Google, IBM, Microsoft etc.
- It aims to provide a universal platform for encoding characters of various languages from the world.
- It assigns a unique Code Point to every character, independent of the CPU architecture/platform, or the underlying software.
- Unicode covers most writing scripts across the world. As of now, over 161 scripts are included in the latest version of Unicode.
- Unicode defines two mapping methods: the Unicode Transformation Format (UTF) encodings, and the Universal Coded Character Set (UCS) encodings.
- Various versions of UTF are UTF-8, UTF-16, UTF-32.
- All UTF encodings map code points to a unique sequence of bytes.

- UTF-8 uses 8-bits or 16-bits or 24-bits or 32-bits for each code point.
- UTF-16 uses 16-bits or 32-bits for each code point.
- UTF-32 uses 32-bits for each code point.

Questions and Answers

Very Short Answer Questions

Q. What is a computer?

A. Computer is an electronic device that takes instructions and data from a user, processes them, and produces some meaningful output as a result.

Q. What is hardware?

A. Hardware are physical components which can be seen and touched (tangible). Examples of hardware include CPU, mouse, keyboard, monitor, motherboard, cables, CPU case, power supply unit, RAM, graphic card, sound card etc.

Q. What is software?

A. Software is a set of instructions, its documentation and data, which are stored digitally on the computer. The software is intangible, i.e cannot be touched. Examples are Microsoft Windows, Linux, Paint, Word, PowerPoint, Photoshop, VLC Media Player, VS Code, Python.

Q. What is an Operating System?

A. An operating system is a system software that acts as an interface between the user and the computer hardware and manages all the resources of a computer.

Q. Give 2 examples of Mobile OS and Desktop OS each.

A. Mobile OS - Android & Apple iOS. Desktop OS - Microsoft Windows, Ubuntu.

Q. What is a mouse?

A. The mouse is an input device responsible for the movement of the cursor on the computer's screen.

Q. What is a keyboard?

A. The keyboard is an input device based on the traditional typewriter, which uses an arrangement of keys. Keyboards come in various layouts and languages.

Q. What is a scanner?

A. Scanner is an input device that optically scans and digitally captures images from physical photographic prints, documents, posters etc. Commonly used office scanners use light beams to scan documents placed on a glass flatbed.

Q. What is a volatile memory?