Number System

The technique to represent and work with numbers is called **number system**. **Decimal number system** is the most common number system. Other popular number systems include **binary number system, octal number system, hexadecimal number system,** etc.

Decimal Number System

Decimal number system is a **base 10** number system having 10 digits from 0 to 9. This means that any numerical quantity can be represented using these 10 digits. Decimal number system is also a **positional value system**.

The weightage of each position can be represented as follows −

10^5	10^4	10^3	10^2	10^1	10^0

Binary Number System

The easiest way to vary instructions through electric signals is two-state system – on and off. On is represented as 1 and off as 0, though 0 is not actually no signal but signal at a lower voltage. The number system having just these two digits – 0 and 1 – is called **binary number system**.

Each binary digit is also called a **bit**. Binary number system is also positional value system, where each digit has a value expressed in powers of 2, as displayed here.

2^5	2^4	2^3	2^2	2^1	2^0

Octal Number System

Octal number system has eight digits – 0, 1, 2, 3, 4, 5, 6 and 7. Octal number system is also a positional value system with where each digit has its value expressed in powers of 8, as shown here −

8^5	8^4	8^3	8^2	8^1	8^0

Hexadecimal Number System

Octal number system has 16 symbols – 0 to 9 and A to F where A is equal to 10, B is equal to 11 and so on till F. Hexadecimal number system is also a positional value system with where each digit has its value expressed in powers of 16, as shown here –

16^5	16^4	16^3	16^2	16^1	16^0

Number System Relationship

The following table depicts the relationship between decimal, binary, octal and hexadecimal number systems.

HEXADECIMAL	DECIMAL	OCTAL	BINARY
0	0	0	0000
1	1	1	0001
2	2	2	0010
3	3	3	0011
4	4	4	0100
5	5	5	0101
6	6	6	0110
7	7	7	0111
8	8	10	1000
9	9	11	1001
A	10	12	1010
B	11	13	1011
C	12	14	1100
D	13	15	1101
E	14	16	1110
F	15	17	1111

Conversion:-

Binary to Octal

An easy way to convert from binary to octal is to group binary digits into sets of three, starting with the least significant (rightmost) digits.

Binary: 11100101 = 11 100 101

011 100 101 Pad the most significant digits with zeros if necessary to complete a group of three.

Then, look up each group in a table:

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

Binary = 011 100 101

Octal = 3 4 5 = 345 oct

Binary to Hexadecimal

An equally easy way to convert from binary to hexadecimal is to group binary digits into sets of four, starting with the least significant (rightmost) digits.

Binary: 11100101 = 1110 0101

Then, look up each group in a table:

Binary:	0000	0001	0010	0011	0100	0101	0110	0111
Hexadecimal:	0	1	2	3	4	5	6	7
Binary:	1000	1001	1010	1011	1100	1101	1110	1111
Hexadecimal:	8	9	A	B	C	D	E	F

Binary = 1110 0101

Hexadecimal = E 5 = E5 hex

Decimal to Binary

Here is an example of using repeated division to convert 1792 decimal to binary:

Decimal Number	Operation	Quotient	Remainder	Binary Result
1792	÷ 2 =	896	0	0

896	÷ 2 =	448	0	00
448	÷ 2 =	224	0	000
224	÷ 2 =	112	0	0000
112	÷ 2 =	56	0	00000
56	÷ 2 =	28	0	000000
28	÷ 2 =	14	0	0000000
14	÷ 2 =	7	0	00000000
7	÷ 2 =	3	1	100000000
3	÷ 2 =	1	1	1100000000
1	÷ 2 =	0	1	11100000000
0	done.			

Decimal to Hexadecimal

Here is an example of using repeated division to convert 1792 decimal to hexadecimal:

Decimal Number	Operation	Quotient	Remainder	Hexadecimal Result
1792	÷ 16 =	112	0	0
112	÷ 16 =	7	0	00
7	÷ 16 =	0	7	700
0	done.			

Decimal Number	Operation	Quotient	Remainder	Hexadecimal Result
48879	÷ 16 =	3054	15	F
3054	÷ 16 =	190	14	EF
190	÷ 16 =	11	14	EEF
11	÷ 16 =	0	11	BEEF
0	done.			

Octal to Binary

Converting from octal to binary is as easy as converting from binary to octal. Simply look up each octal digit to obtain the equivalent group of three binary digits.

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

Octal = 3 4 5
Binary = 011 100 101 = 011100101 binary

Octal to Hexadecimal

When converting from octal to hexadecimal, it is often easier to first convert the octal number into binary and then from binary into hexadecimal. For example, to convert 345 octal into hex:

(from the previous example)

Octal = 3 4 5

Binary = 011 100 101 = 011100101 binary

Drop any leading zeros or pad with leading zeros to get groups of four binary digits (bits):
Binary 011100101 = 1110 0101

Then, look up the groups in a table to convert to hexadecimal digits.

Binary:	0000	0001	0010	0011	0100	0101	0110	0111
Hexadecimal:	0	1	2	3	4	5	6	7
Binary:	1000	1001	1010	1011	1100	1101	1110	1111
Hexadecimal:	8	9	A	B	C	D	E	F

Binary = 1110 0101
Hexadecimal = E 5 = E5 hex

Therefore, through a two-step conversion process, octal 345 equals binary 011100101 equals hexadecimal E5.

Decimal to Binary

Decimal numbers can be converted to binary by repeated division of the number by 2 while recording the remainder. Let's take an example to see how this happens.

		Remainder	
2	43		
2	21	1	MSB ↑
2	10	1	
2	5	0	
2	2	1	
2	1	0	
	0	1	LSB

The remainders are to be read from bottom to top to obtain the binary equivalent.

$43_{10} = 101011_2$

Decimal to Octal

Decimal numbers can be converted to octal by repeated division of the number by 8 while recording the remainder. Let's take an example to see how this happens.

		Remainder	
8	473		
8	59	1	MSD
8	7	3	↑
	0	7	LSD

Reading the remainders from bottom to top,

$473_{10} = 731_{8}$

Decimal to Hexadecimal

Decimal numbers can be converted to octal by repeated division of the number by 16 while recording the remainder. Let's take an example to see how this happens.

		Remainder	
16	423		
16	26	7	↑
16	1	A	
	0	1	

Reading the remainders from bottom to top we get,

$423_{10} = 1A7_{16}$

Binary Arithmetic

Binary Addition

It is a key for binary subtraction, multiplication, division. There are four rules of binary addition.

Case	A	+	B	Sum	Carry
1	0	+	0	0	0
2	0	+	1	1	0
3	1	+	0	1	0
4	1	+	1	0	1

In fourth case, a binary addition is creating a sum of (1 + 1 = 10) i.e. 0 is written in the given column and a carry of 1 over to the next column.

Example − Addition

0011010 + 001100 = 00100110

```
  1 1          carry
 0 0 1 1 0 1 0  = 26₁₀
+0 0 0 1 1 0 0  = 12₁₀
---------------
 0 1 0 0 1 1 0  = 38₁₀
```

Binary Subtraction

Subtraction and Borrow, these two words will be used very frequently for the binary subtraction. There are four rules of binary subtraction.

Case	A	-	B	Subtract	Borrow
1	0	-	0	0	0
2	1	-	0	1	0
3	1	-	1	0	0
4	0	-	1	0	1

Example − Subtraction

0011010 - 001100 = 00001110

```
     1 1        borrow
 0 0 1 1 0 1 0  = 26₁₀
-0 0 0 1 1 0 0  = 12₁₀
---------------
 0 0 0 1 1 1 0  = 14₁₀
```

Binary Multiplication

Binary multiplication is similar to decimal multiplication. It is simpler than decimal multiplication because only 0s and 1s are involved. There are four rules of binary multiplication.

Case	A	x	B	Multiplication
1	0	x	0	0
2	0	x	1	0
3	1	x	0	0
4	1	x	1	1

Example − Multiplication

Example:

0011010 x 001100 = 100111000

```
          0011010    = 26₁₀
         x0001100    = 12₁₀
     ------------
          0000000
         0000000
        0011010
       0011010
     ------------
       0100111000    = 312₁₀
```

Binary Division

Binary division is similar to decimal division. It is called as the long division procedure.

Example − Division

101010 / 000110 = 000111

```
               111      = 7₁₀
        ----------
000110 ) 101010        = 42₁₀
        -110            = 6₁₀
        ----
          1001
          -110
          ----
           110
          -110
          ----
             0
```