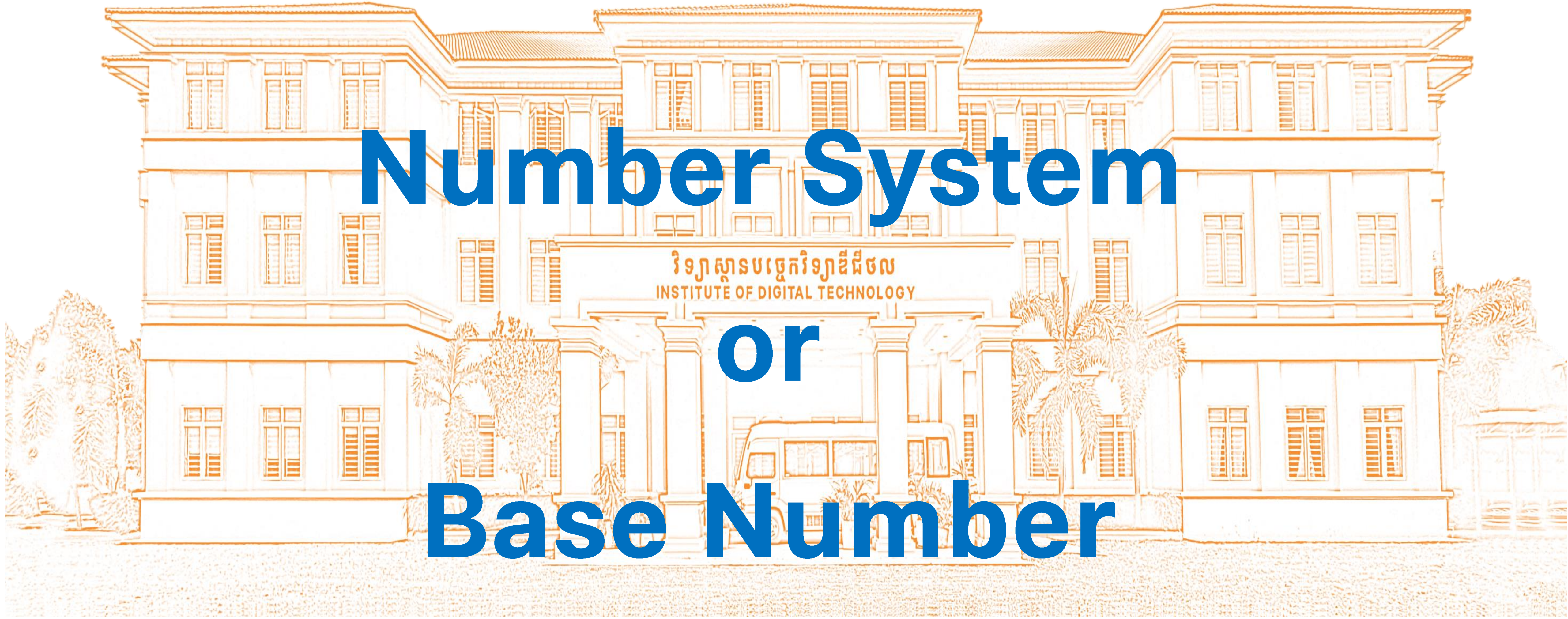
CADT
IDT វិទ្យាស្ថានបច្ចេកវិទ្យាឌីជីថល
Institute of Digital Technology

Department of Foundation Year

Number System

or

Base Number



NGETH YOUDARITH
youdarith.ngeth@cadt.edu.kh
(+855) 12 600 012

History of Number System

The history of number systems dates back thousands of years, as humans have been counting and representing numbers since ancient times. Here's a brief overview of the development of number systems:

1. Prehistoric Counting: The earliest form of counting can be traced back to prehistoric times, where humans used simple tally marks or objects like stones to keep track of quantities.

History of Number System

2.Ancient Number Systems:

Babylonian Number System: The Babylonians, around 1800 BCE, used a base-60 number system known as the sexagesimal system. It consisted of symbols for numbers 1 to 59 and a special symbol for zero.

Egyptian Number System: The ancient Egyptians used a decimal system around 3000 BCE. They represented numbers using hieroglyphic symbols for powers of ten and specific symbols for the digits.

Roman Numeral System: The Romans developed a numeral system based on specific letters representing values. It was not a positional system and lacked a symbol for zero.

History of Number System

3.Indian Number System: The Indian mathematician Aryabhata introduced the Indian number system (also known as the Hindu-Arabic numeral system) around the 5th century CE. It was a decimal positional system with ten digits (0-9) and a symbol for zero, which revolutionized mathematics and facilitated complex calculations.

4.Development of Notation Systems:

Brahmi Numerals: The ancient Indians developed the Brahmi numerals, which were the precursors to modern Indian numerals. They were used around the 3rd century BCE.

Eastern Arabic Numerals: The Indian number system reached the Arab world around the 9th century CE. Arab mathematicians further developed the system and introduced it to Europe, where it became known as the "Arabic numerals."

History of Number System

5.Introduction of Zero: The concept of zero as a placeholder and as a numerical value was developed in various ancient civilizations, including India and the Maya. The Indian mathematician Brahmagupta explicitly defined zero as a number in the 7th century CE.

6.Positional Number Systems: The modern decimal system, based on the Indian number system, became widely adopted in Europe during the 12th century. It is a positional system where the value of a digit depends on its position within the number.

History of Number System

7.Expansion to Other Bases: In addition to the decimal system, other number bases have been used throughout history. For example:

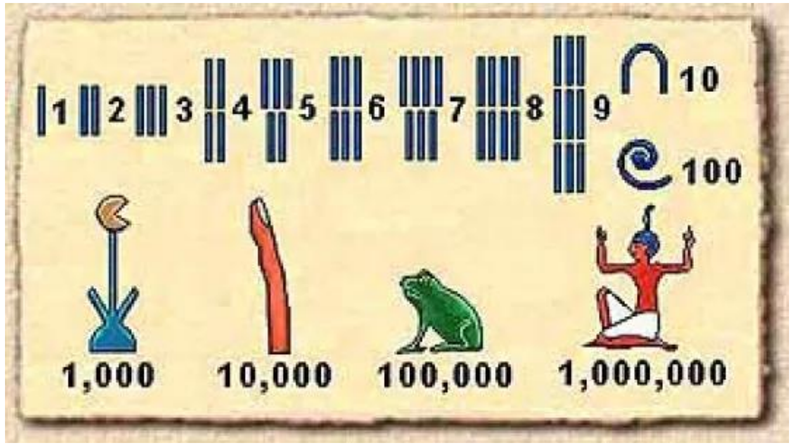
- ❖ Binary System: The binary system, base-2, was used by the ancient Chinese and was later formalized by the German mathematician Gottfried Wilhelm Leibniz in the 17th century. It is fundamental in modern computing.
- ❖ Hexadecimal System: The hexadecimal system, base-16, gained popularity in the mid-20th century for its use in representing large binary numbers more concisely. It is commonly used in computer science and programming.

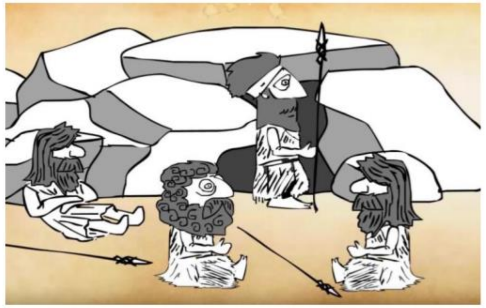
History of Number System

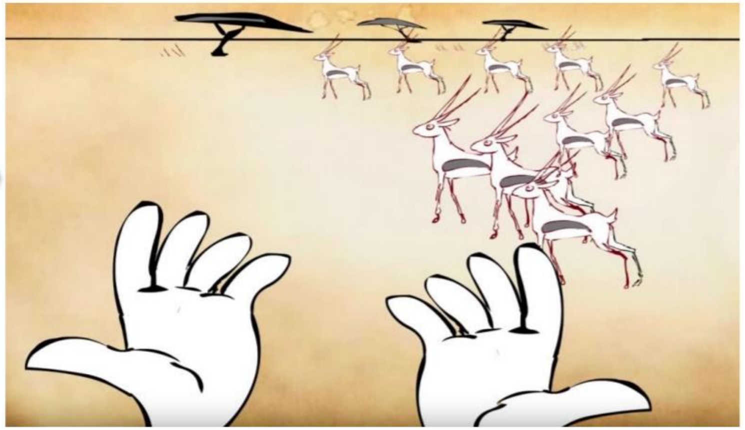
The history of number systems reflects the evolution of human understanding and the development of mathematical concepts. The introduction of positional systems, the inclusion of zero, and the adoption of various bases have played a significant role in advancing mathematics, science, and technology.

Number System

Introduction of Number System or Base-Number

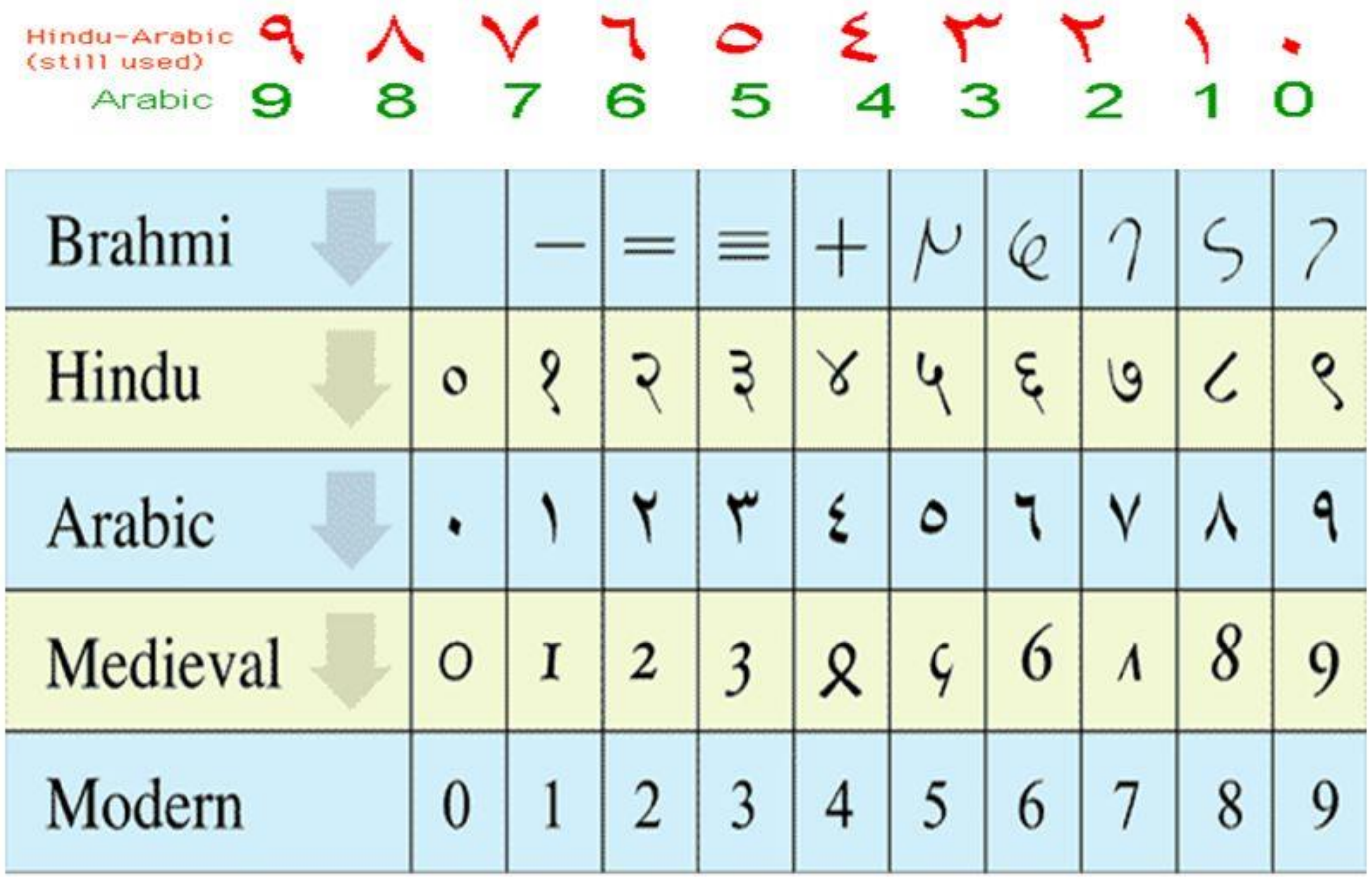




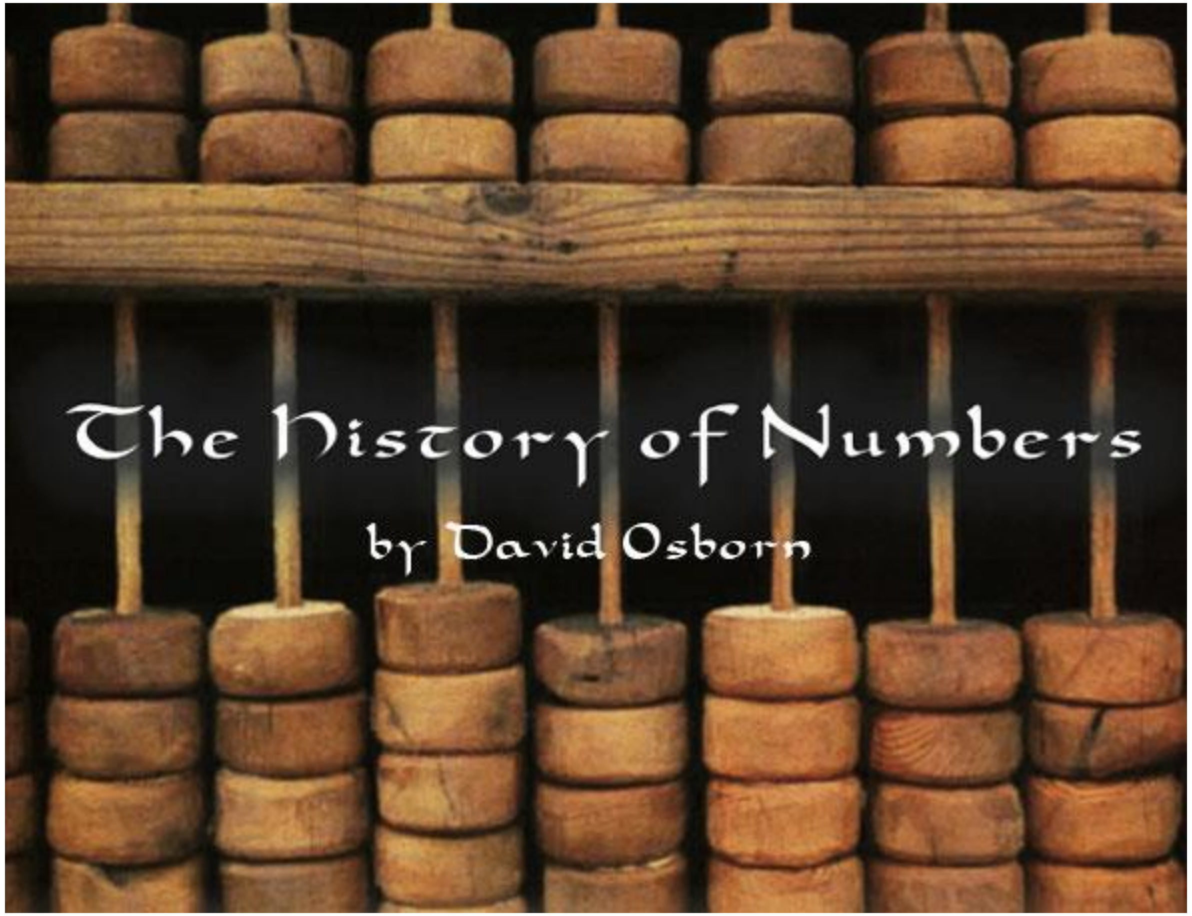


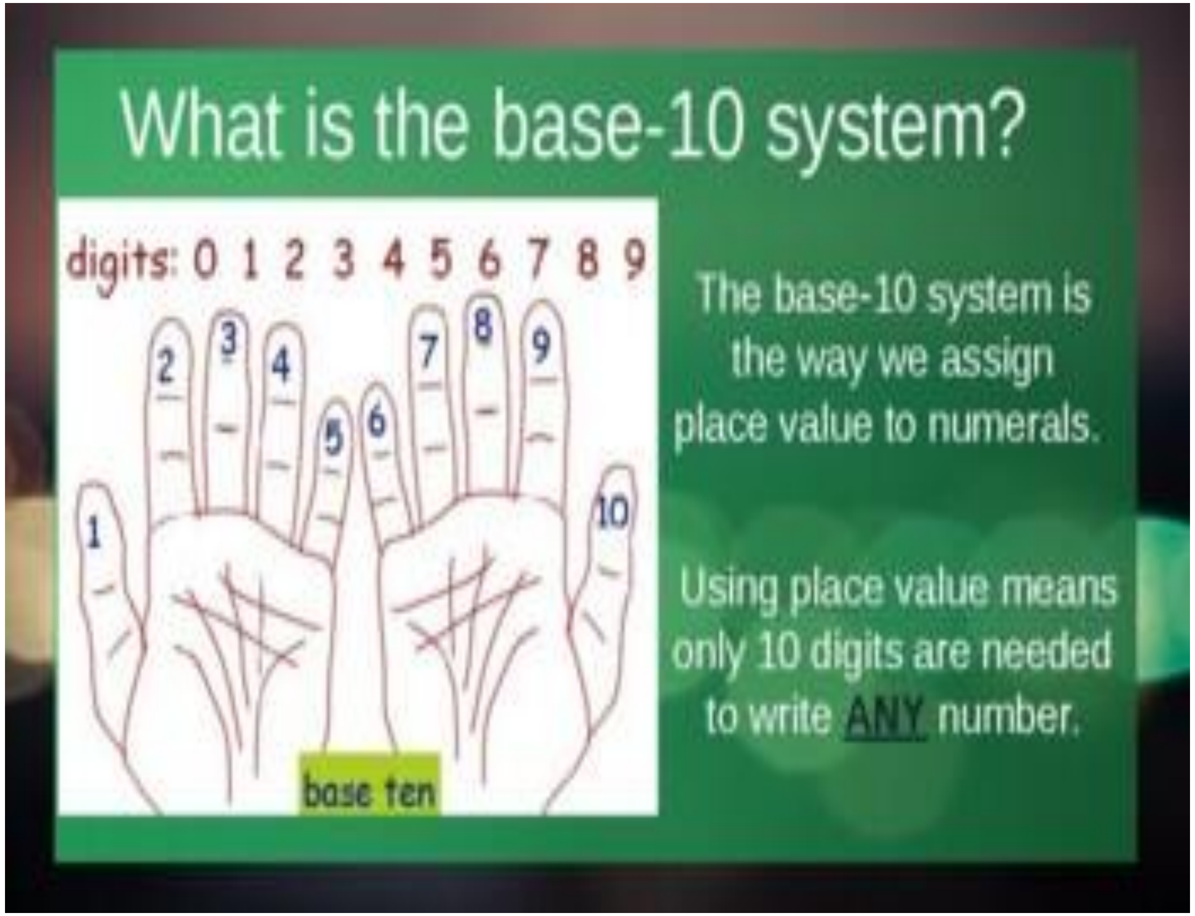
Number System

The numbers we use today, called HinduArabic numerals, were first developed by Indian mathematicians.



Number System





Number System

What is Number System in Math?

A number system is defined as a system of writing to express numbers. It is the mathematical notation for representing numbers of a given set by using digits or other symbols in a consistent manner. It provides a unique representation of every number and represents the arithmetic and algebraic structure of the figures. It also allows us to operate arithmetic operations like addition, subtraction and division.

The value of any digit in a number can be determined by:

The digit

Its position in the number

The base of the number system

Number System

why we study number system for computer science, network telecom and ecommerce?

Studying the number system is important in the fields of computer science, network telecom, and e-commerce for several reasons:

1. Representation of Information: Numbers are used to represent and manipulate various types of information in these fields. Understanding number systems allows us to represent data such as text, images, sounds, and other multimedia in a digital format.

2. Binary Representation: Computers and digital systems use the binary number system (base-2) to process and store information. Binary digits (bits) are the building blocks of digital data, and knowledge of binary operations is essential for understanding how computers and digital systems operate.

Number System

3.Data Storage and Compression: Knowledge of number systems helps in understanding how data is stored and compressed in computer systems. Understanding data compression techniques allows for efficient storage and transmission of data, which is crucial in network telecom and e-commerce applications where large amounts of data are processed.

4.Data Encryption and Security: Number systems play a significant role in cryptography and data security. Encryption algorithms, such as the widely used RSA algorithm, are based on mathematical operations performed on numbers. Understanding number systems helps in comprehending encryption techniques and implementing secure communication systems.

Number System

5.Network Addressing: In network telecom, understanding number systems is vital for working with IP addressing and subnetting. Internet Protocol (IP) addresses are represented as a series of numbers, and subnetting involves dividing a network into smaller subnetworks using binary operations.

6.Algorithms and Data Structures: Number systems form the foundation for many algorithms and data structures used in computer science. Concepts such as number sorting, searching, and arithmetic operations heavily rely on an understanding of number systems.

Number System

7.E-commerce Transactions: In e-commerce, numbers are used extensively for financial transactions. Understanding number systems, decimal representation, and arithmetic operations is crucial for accurate and secure monetary transactions.

Overall, studying number systems provides a fundamental understanding of how information is represented, processed, and transmitted in the digital world. It is essential for professionals in computer science, network telecom, and e-commerce to have a strong grasp of number systems to work effectively in their respective fields.

Number System

Types of Number System

There are various types of number system in mathematics. The four most common number system types are:

Decimal number system (Base- 10)

Binary number system (Base- 2)

Octal number system (Base-8)

Hexadecimal number system (Base- 16)

Base 32 Number system

Base 64 Number System

Number System

Decimal Number System (Base 10 Number System)

Decimal number system has base 10 because it uses ten digits from 0 to 9. In the decimal number system, the positions successive to the left of the decimal point represent units, tens, hundreds, thousands and so on. This system is expressed in decimal numbers.

Every position shows a particular power of the base (10). For example, the decimal number 1457 consists of the digit 7 in the units position, 5 in the tens place, 4 in the hundreds position, and 1 in the thousands place whose value can be written as

$$(1 \times 10^3) + (4 \times 10^2) + (5 \times 10^1) + (7 \times 10^0)$$

$$(1 \times 1000) + (4 \times 100) + (5 \times 10) + (7 \times 1)$$

$$1000 + 400 + 50 + 7$$

$$1457$$

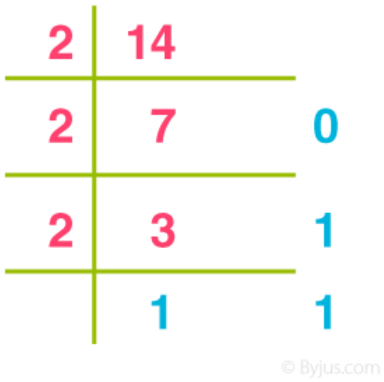
Number System

Binary Number System (Base 2 Number System)

The base 2 number system is also known as the Binary number system wherein, only two binary digits exist, i.e., 0 and 1. Specifically, the usual base-2 is a radix of 2. The figures described under this system are known as binary numbers which are the combination of 0 and 1. For example, 110101 is a binary number.

We can convert any system into binary and vice versa.

Example : Write $(14)_{10}$ as a binary number.



$\therefore (14)_{10} = 1110_2$

Number System

Octal Number System (Base 8 Number System)

In the octal number system, the base is 8 and it uses numbers from 0 to 7 to represent numbers. Octal numbers are commonly used in computer applications. Converting an octal number to decimal is the same as decimal conversion and is explained below using an example.

Example: Convert 215_8 into decimal.

Solution:

$$215_8 = 2 \times 8^2 + 1 \times 8^1 + 5 \times 8^0$$

= 2 × 64 + 1 × 8 + 5 × 1

= 128 + 8 + 5

$$= 141_{10}$$

Number System

Hexadecimal Number System (Base 16 Number System)

In the hexadecimal system, numbers are written or represented with base 16. In the hex system, the numbers are first represented just like in decimal system, i.e. from 0 to 9. Then, the numbers are represented using the alphabets from A to F. The below-given table shows the representation of numbers in the hexadecimal number system.

Hexadecimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Deicimal	Hexadecimal	Binary	Octal
0	0	0	0
1	1	1	1
2	2	10	2
3	3	11	3
4	4	100	4
5	5	101	5
6	6	110	6
7	7	111	7
8	8	1000	10
9	9	1001	11
10	A	1010	12
11	B	1011	13
12	C	1101	14
13	D	1101	15
14	E	1110	16
15	F	1111	17

Number System

Number System Chart

In the number system chart, the base values and the digits of different number system can be found. Below is the chart of the numeral system.

Number System	Base value	Set of digits	Example
Base 3	3	0, 1, 2	$(122)_3$
Base 4	4	0, 1, 2, 3	$(123)_4$
Base 5	5	0, 1, 2, 3, 4	$(243)_5$
Base 6	6	0, 1, 2, 3, 4, 5	$(225)_6$
Base 7	7	0, 1, 2, 3, 4, 5, 6	$(1205)_7$
Base 8	8	0, 1, 2, 3, 4, 5, 6, 7	$(105)_8$
Base 9	9	0, 1, 2, 3, 4, 5, 6, 7, 8	$(25)_9$
Base 10	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	$(1125)_{10}$

Number System

Base 32 Number System

In the Base 32 Number System, numbers are written or represented with base 32. In Base 32 Number System, the numbers are first represented just like in alphabets A-Z and number 2-7. The below-given table shows the representation of numbers in the Base 32 number system.

Value	Symbol		Value	Symbol		Value	Symbol		Value	Symbol
0	A		9	J		18	S		27	3
1	B		10	K		19	T		28	4
2	C		11	L		20	U		29	5
3	D		12	M		21	V		30	6
4	E		13	N		22	W		31	7
5	F		14	O		23	X			
6	G		15	P		24	Y			
7	H		16	Q		25	Z			
8	I		17	R		26	2			

Number System

Base 64 Number System

In the Base 64 Number System, numbers are written or represented with base 64. In Base 64 Number System, the numbers are first represented just like in alphabets A-Z and small alphabets a-z and number 0-9 include operation + and / . The below-given table shows the representation of numbers in the Base 64 number system.

Value	Char		Value	Char		Value	Char		Value	Char
0	A		16	Q		32	g		48	w
1	B		17	R		33	h		49	x
2	C		18	S		34	i		50	y
3	D		19	T		35	j		51	z
4	E		20	U		36	k		52	0
5	F		21	V		37	l		53	1
6	G		22	W		38	m		54	2
7	H		23	X		39	n		55	3
8	I		24	Y		40	o		56	4
9	J		25	Z		41	p		57	5
10	K		26	a		42	q		58	6
11	L		27	b		43	r		59	7
12	M		28	c		44	s		60	8
13	N		29	d		45	t		61	9
14	O		30	e		46	u		62	+
15	P		31	f		47	v		63	/

Conversion

Number System Conversion

Numbers can be represented in any of the number system categories like binary, decimal, hex, etc. Also, any number which is represented in any of the number system types can be easily converted to other. Check the detailed lesson on the conversions of number systems to learn how to convert numbers in decimal to binary and vice versa, hexadecimal to binary and vice versa, and octal to binary and vice versa using various examples.

Conversion

How to convert decimal to binary

1. Divide the number by 2.

2. Get the integer quotient for the next iteration.

3. Get the remainder for the binary digit.

4. Repeat the steps until the quotient is equal to 0.

Conversion

Convert Decimal to Octal with Steps

Follow the steps given below to learn the decimal to octal conversion:

1. Write the given decimal number
2. If the given decimal number is less than 8 the octal number is the same.
3. If the decimal number is greater than 7 then divide the number by 8.
4. Note the remainder, we get after division
5. Repeat step 3 and 4 with the quotient till it is less than 8
6. Now, write the remainders in reverse order (bottom to top)
7. The resultant is the equivalent octal number to the given decimal number.

Conversion

How to convert from decimal to Hexadecimal

1. Divide the decimal number by 16. Treat the division as an integer division.

2. Write down the remainder (in hexadecimal).

4. Divide the result again by 16. Treat the division as an integer division.

5. Repeat step 2 and 3 until result is 0.

6. The hex value is the digit sequence of the remainders from the last to first.

Conversion

Decimal to Other Base System Steps

Step 1 – Divide the decimal number to be converted by the value of the new base.

Step 2 – Get the remainder from Step 1 as the rightmost digit (least significant digit) of new base number.

Step 3 – Divide the quotient of the previous divide by the new base.

Step 4 – Record the remainder from Step 3 as the next digit (to the left) of the new base number.

Repeat Steps 3 and 4, getting remainders from right to left, until the quotient becomes zero in Step 3.

The last remainder thus obtained will be the Most Significant Digit (MSD) of the new base number.

Conversion

Solved Examples

Q.1: Convert $(1056)_{16}$ to octal number.

Solution: Given, 1056_{16} is an hex number.

First we need to convert the given hexadecimal number into decimal number

$$(1056)_{16}$$

$$= 1 \times 16^3 + 0 \times 16^2 + 5 \times 16^1 + 6 \times 16^0$$

$$= 4096 + 0 + 80 + 6$$

$$= (4182)_{10}$$

Conversion

Now we will convert this decimal number to required octal number by repetitively dividing by 8.

8	4182	Remainder
8	522	6
8	65	2
8	8	1
8	1	0
	0	1

Therefore, taking the value of remainder from bottom to top, we get;

$(4182)_{10} = (10126)_8$

Therefore, $(1056)_{16} = (10126)_8$

Conversion

2: Convert $(1001001100)_2$ to decimal number.

Solution: $(1001001100)_2$

$= 1 \times 2^9 + 0 \times 2^8 + 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$

$= 512 + 64 + 8 + 4$

$= (588)_{10}$

Conversion

Convert 10101_2 into octal number.

Solution: Given,

10101_2 is the binary number

We can write the given binary number as:

010 101

Now as we know, in octal number system,

$010 \rightarrow 2$

$101 \rightarrow 5$

Therefore, the required octal number is 25_8

Conversion

4: Convert hexadecimal 2C to decimal number.

Solution: We need to convert $2C_{16}$ into binary number first.

$2C \rightarrow 00101100$

Now convert 00101100_2 into a decimal number.

$101100 = 1\times2^5+1\times2^3+1\times2^2$

$=32+8+4$

$=44$

Conversion

5: Convert $(10111)_2$ to Octal number.

Conversion

6. Convert $(11010011)_2$ to Hexadecimal number.

Conversion

7. Convert $(2112)_8$ to decimal number.

Conversion

8. Convert $(4141)_8$ to Hexadecimal number.

Conversion

9. Convert $(4141)_8$ to binary number.

Conversion

9. Convert $(2023)_{16}$ to binary number.

Conversion

10. Convert $(ABC)_{16}$ to binary and Octal number.

Conversion

11. Convert $(25)_{10}$ to Binary number.

Conversion

11. Convert $(225)_{10}$ to Binary octal and hexadecimal number.

Conversion

13. Convert $(5BA2)_{16}$ to octal number.

Conversion

14. Convert $(BCD)_{64}$ to Binary octal and hexadecimal number.

Conversion

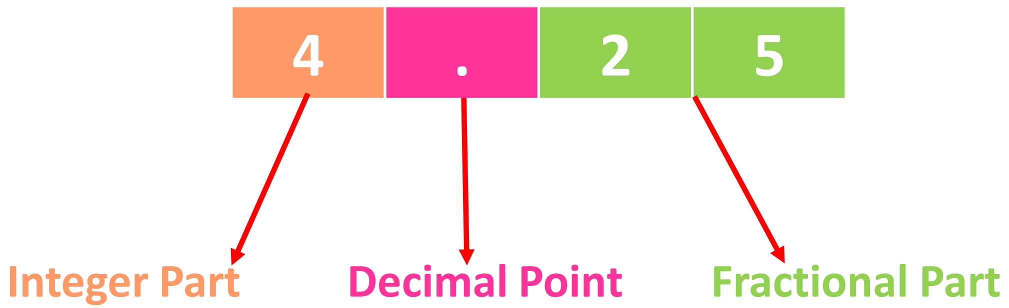
How to convert floating numbers into binary numbers ?

Let's take an example Convert 4.25 to binary numbers

Where,

4 is an integral part.

0.25 is a fractional part.



Conversion

Integral Part (4)

To convert an integral part into binary, just follow the previously discussed method.

Binary Number System

Using that method, we can represent 4 as (100) 2.

Conversion

Fractional part (0.25)

To convert the fractional part to binary, multiply fractional part with 2 and take the one bit which appears before the decimal point.

Follow the same procedure with after the decimal point (.) part until it becomes 1.0

Like,

0.25 * 2 =0.50 //take 0 and move 0.50 to next step

0.50 * 2 =1.00 //take 1 and stop the process

0.25 = (01) 2

Combining both integral and fractional,

4.25 = (100.01) 2

Conversion

Example Convert 2.33 to binary

Integral Part

2 = (10) 2

Fractional Part

0.33 * 2 =>0.66 // take 0 and move 0.66 to next step
0.66 * 2 =>1.32 // take 1 and move 1.32 to next step
1.32 * 2 =>0.64 // take 0 and move 0.64 to next step
0.64 * 2 =>1.28 // take 1 and move 1.28 to next step
1.28 * 2 =>0.56 // take 0 and move 0.56 to next step
0.56 * 2 =>1.12 // take 1 and move 1.12 to next step
1.12 * 2 =>0.24 // take 0 and move 0.24 to next step
0.24 * 2 =>0.48 // take 0 and move 0.48 to next step
0.48 * 2 =>0.96 // take 0 and move 0.96 to next step
0.96 * 2 =>1.92 // take 1 and move 1.92 to next step
1.92 * 2 =>1.84 // take 1 and move 1.84 to next step
1.84 * 2 =>1.68 // take 1 and move 1.68 to next step
1.68 * 2 =>1.36 // take 0 and move 1.36 to next step
1.36 * 2 =>0.72 // take 0 and move 0.72 to next step
0.72 * 2 =>1.44 // take 1 and move 1.44 to next step
1.44 * 2 =>0.88 // take 0 and move 0.88 to next step
0.88 * 2 =>1.76 // take 1 and move 1.76 to next step
1.76 * 2 =>1.52 // take 1 and move 1.52 to next step
1.52 * 2 =>1.04 // take 1 and move 1.04 to next step
1.04 * 2 =>0.08 // take 1 and move 0.08 to next step
0.08 * 2 =>0.16 // take 1 and move 0.16 to next step
0.16 * 2 =>0.32 // take 1 and move 0.32 to next step
0.32 * 2 =>0.64 // take 1 and move 0.64 to next step

Conversion

Here, the fractional part 0.32 which is repeating again.

And Some fractional part numbers will not end up with 1.0 with the above method.

In floating number storage, the computer will allocate 23 bits for the fractional part. So, it's enough to do the above method at max 23 times.

2.33 = (10.010101000111001011) 2

Conversion

Example Convert 168.65625 to Octal and Hexadecimal

Conversion

Example Convert $(176.422)_8$ to Decimal and Hexadecimal

Conversion

Example Convert $(215.168)_{16}$ to Decimal and Octal

Binary Addition Rules

Addition	sum	carry
0 + 0	0	0
0 + 1	1	0
1 + 0	1	0
1 + 1	0	1

$$
\begin{array}{r}
1011.01 \\
\times \quad 110.1 \\
\hline
101101 \\
0000000 \\
10110100 \\
101101000 \\
\hline
1001001.001
\end{array}
$$

$$
\begin{array}{r}
110 \\
11 \overline{)\,10010} \\
11 \\
\hline
11 \\
11 \\
\hline
00 \\
0 \\
\hline
0
\end{array}
$$