Digital Data 1/15/25

I) Numbers

II) A different paradigm: hexadecimals

Decimal: $N = a\,10^3 + b\,10^2 + c\,10 + d$

Binary $N = a\,2^3 + b\,2^2 + c\,2 + d$

Hexadecimal $N = a\,16^3 + b\,16^2 + c\,16 + d$

The use of hexadecimal numbers:

binary number N

$$N = (\underbrace{\boxed{1011}}_{B}\ \underbrace{1000}_{8}\ \underbrace{1001}_{9}\ \underbrace{0001}_{1})_2$$

$$N = (B891)_{16}$$

$$= \#B891$$

Reversely,

Take $N = \# 8FA$

8	F	A

$N = (1000 \quad 1111 \quad 1010)_2$

III) Texts

	Conversion
a	$\emptyset$
b	1
c	$1\emptyset$
d	11

$\emptyset \; \emptyset$ a a

$\boxed{1 \; 1 \; \emptyset}$ ambiguous $\rightarrow$ $\underset{b}{1} \; \underset{c}{1\emptyset}$

$\searrow$ $\underset{d}{11} \; \underset{a}{\emptyset}$

This representation

a	0
b	1
c	10
d	11

is ambiguous.

Instead, I give exactly 2 bits to each letter:

a	00
b	01
c	10
d	11

$\boxed{10}\boxed{11}$

For a larger alphabet (English)

ASCII: $\underbrace{\text{8 bits}}_{\text{256 types of characters}}$

today, ASCII is not enough as it only defines 256 characters.

UNICODE : 156,000 characters.

	# of characters	# of bits/character
ASCII	256	8
UNICODE	156,000	18

N such that

$$2^{N-1} \le 156,000 < 2^{N}$$