Session 4

Friday, 11 July 2025 9:53 AM

Operatos	Type of Operatos	Operation Type
++, --	Increments/Decrements Operators	Unary Operator
+, -, *, /, %	Arithmetic Operators	Binary Operator
<, <=, >, >=, ==, !=	Relational Operators	
&&, |, !	Logical Operators	
&, |, <<, >>, ~, ^	Bitwise Operators	
=, +=, -=, *=, /=, %=	Assignment Operators	
sizeof () , & *	Special Operaots	
?:	Ternary or Conditional Operator	Ternary Operator

if / else / while / do while / for

Arithemetic / Relational operator

+, -, *, /, % <, >, <=, >=, !=, ==

Number System :-

Decimal Number System :-

(10 digits)

0 1 2 3 4 5 6 7 8 9

Decimal Number System

→ Base - (10)

→ radix

↳ How many digits are in the Particular Number System.

(5557) → five thousand five hundred fifty seven

ones

10's

100's

1000's

Any Number has Power Zero .. than the result is One $[X]^0 = 1$

Base → (10)

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

$5 \times 1000 + 5 \times 100 + 5 \times 10 + 7 \times 1$

↳ 1000's ↳ 100's ↳ 10's ↳ ones

$5000 + 500 + 50 + 7$

5557

Decimal Number System → (10) → 0 – 9

Octal Number System → (8) → 0 – 7

Hexa Number System → (16) → 0 – 15

Binary Number System → (2) → 0 – 1

✓ Number System Having Base → (5) → 0 – 4

Decimal Number System:

0)+1
1)+1
2)+1
3)+1
4)+1
5)+1
6)+1
7)+1
8)+1
[9])+1
10 → (Ten)

Carry Sum

$$\begin{array}{r} 1 \\ +1 \\ \hline 10 \end{array} \qquad \begin{array}{r} 0 \\ +0 \\ \hline 0 \end{array} \qquad \begin{array}{r} 0 \\ +1 \\ \hline 1 \end{array} \qquad \begin{array}{r} 1 \\ +0 \\ \hline 1 \end{array}$$

$$\begin{array}{r} 9 \\ +1 \\ \hline 10 \end{array} \qquad \begin{array}{r} 11 \\ +1 \\ \hline 100 \end{array} \qquad \begin{array}{r} 111 \\ +1 \\ \hline 1000 \end{array}$$

Binary Number → Machine

(0,1) → (2) Base

1 is Maximum digit in this Number System

0 – 0000)+1
1 – 0001)+1
2 – 0010)+1
3 – 0011)+1
4 – 0100)+1
5 – 0101)+1
6 – 0110)+1
7 – 0111)+1
8 – 1000)+1
9 – [1001]

$(1001)_2 \rightarrow (?)_{10}$

1 0 0 1

$1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$

$1 \times 8 + 0 \times 4 + 0 \times 2 + 1 \times 1$

$8 + 0 + 0 + 1 \Rightarrow (9)$

If you add one in Maximum digit of Any Number System You Always Get Sum Zero & Carry one.

0
1
9

$$\begin{array}{r} 9 \\ +1 \\ \hline 10 \end{array}$$

Decimal		Binary
0	–	0000
1	–	0001
2	–	0010
3	–	0011
4	–	0100
5	–	0101
6	–	0110
7	–	0111
8	–	1000
9	–	1001

{0, 1}

$$\begin{array}{r} (1) \\ +1 \\ \hline 10 \end{array}$$

Logical Operator :- &&, ||, !
AND, OR, NOT

(Binary Operator)

Truth Table for AND

Zero → False
NonZero → True
Binary

X	Y	Z / output
0	0	0 ✓
0	1	0 ✓

If Both the Input is high (NonZero) then only the output of logical And is High otherwise it gives Low output (Zero)

1×0×1 => 1

OR (Logical OR) = ||

if any of the Input is high (Nonzero = True) than the output is high otherwise Zero (low) false

0 – false
Non Zero = True

1+1 => 1
0 + 1
0+0+0
0

Truth Table is As follows

X	Y	output (2)
0	0	0
0	1	1
1	0	1
1	1	1

Not – (!) Logical Not :- it alters the Input

Truth Table.

X	Y
0	1
1	0

The logical operators are Use to Combine the Condition (2 or More)

```
if((age > 18) && (age < 60))
{ print (" Hey iM a Valid Driver");
```

X	
0	1
1	0

Unsigned and Signed

Compiler ← 5 && 0 || = 0 || 0

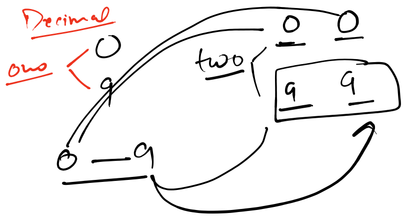
false = 0
True = Non zero

false 0 || 1 = 0 || 1 || 0
True || 0 ⇒ 1
True || True ⇒ True

Signed : or Unsigned :-

Type	Size (bits)	Size (bytes)	Range
char	8	1	-128 to 127
unsigned char	8	1	0 to 255
int	16	2	-2^{15} to 2^{15}-1
unsigned int	16	2	0 to 2^{16}-1
short int	8	1	-128 to 127
unsigned short int	8	1	0 to 255
long int	32	4	-2^{31} to 2^{31}-1
unsigned long int	32	4	0 to 2^{32}-1
float	32	4	3.4E-38 to 3.4E+38
double	64	8	1.7E-308 to 1.7E+308
long double	80	10	3.4E-4932 to 1.1E+4932

-127 to +127



Char ?? ;

8 bit

0 ← 0 0 0 0 0 0 0 0

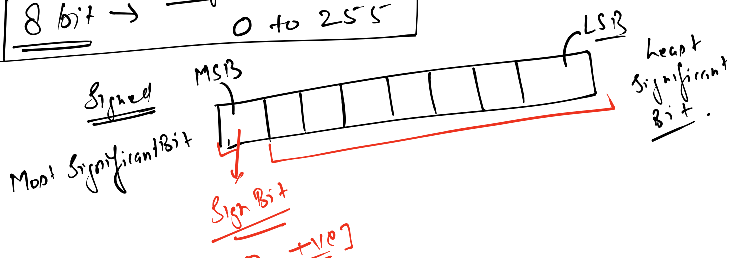
← 1 1 1 1 1 1 1 1 → (2)

$1\times2^7+1\times2^6+1\times2^5+1\times2^4+1\times2^3+1\times2^2+1\times2^1+1\times2^0$

$128+64+32+16+8+4+2+1$

⇒ 255

8 bit → Unsigned 0 to 255



MSB, LSB

+ve 0 1

-ve

0 — 000, 111 ($2^2+2^1+2^0$ ⇒ 7)

1 — 000, 111

+0 to +7

-0 to -7

4 bit

Sign Magnitude Representation

n is Number of Bits

Bit

-7 to $+7$

$-2^{n-1}-1$ to $+2^{n-1}-1$

$-2^{4-1}-1$ to $2^{4-1}-1$

$-(2^3-1)$ to 2^3-1

-7 to $+7$

+ve

-ve

-8 -7 -6 -5 -4 -3 -2 -1 -0 $+0$ $+1$ $+2$ $+3$ $+4$ $+5$ $+6$ $+7$

-8 -7 -6 -5 -4 -3 -2 -1 $+0$ $+1$ $+2$ $+3$ $+4$ $+5$ $+6$ $+7$ -8

-8 to $+7$

-2^{n-1} to $+2^{n-1}-1$

-2^{n-1} to $+2^{n-1}-1$

$-(2^{4-1}-1)$ $-(2^3-1)$

-7

2's Complement format :-

$+7 \Rightarrow$ 0 0 0 0 0 1 1 1

2's Complement

1's Complement $00000111 \rightarrow 11111000$

2's Complement $\rightarrow$ 1's Complement $+1$

1's $00000111 \rightleftarrows 11111000$

$+1$

2's Complement $11111001 \rightarrow 11111001$

1's $01010100 \rightarrow 10101011$

$+1$

2's Complement $10101100 \rightarrow 10101100$

ones
0101011100100
1010100011011
+1
+1
1010100111100

101010011100

0100110011
1011001101 → 2's

System Store the Number in 2's which also a weighted System sign Bit is always −ve

+7
−7

+7 00000111
−7 11111001

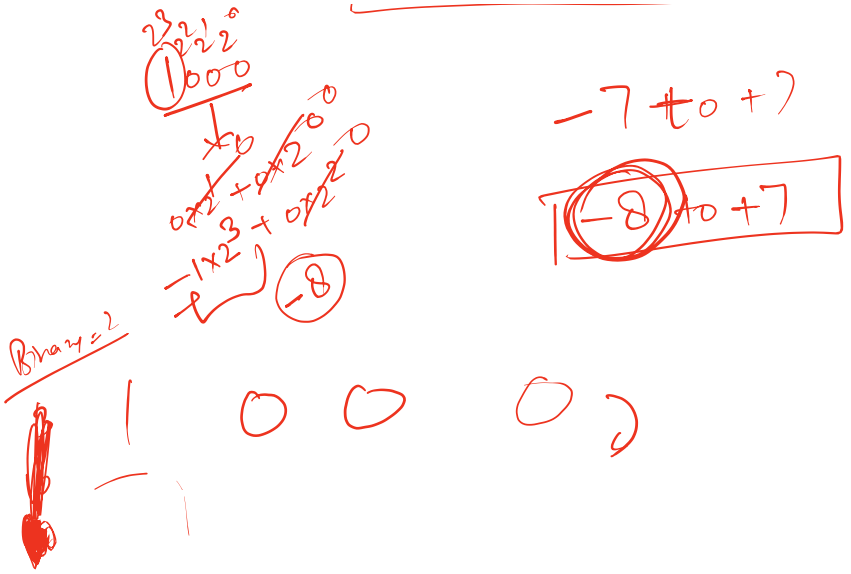
11111001
$-1\times2^7 + 1\times2^6 + 1\times2^5 + 1\times2^4 + 1\times2^3 + 0\times2^2 + 0\times2^1 + 1\times2^0$
$-128 + 64 + 32 + 16 + 8 + 0 + 0 + 1$
$-128 + 121$
−7

+7 $-2^7\times0 + 2^6\times0 + 2^5\times0\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$
00000111
$2^2 + 2^1 + 2^0 \Rightarrow +7$

4 Bit MSB

+0	0000	0000 −0
+1	0001	1111
+2	0010	1110
+3	0011	1101
+4	0100	1000
+5	0101	1011
+6	0110	1010
+7	0111	1001

1000


2³ 2² 2¹ 2⁰
1 0 0 0
1×2³ + 0×2² + 0×2¹ + 0×2⁰
-1×2³ + 0×2² + 0×2¹ + 0×2⁰
-8
-7 to 0 to +7
-8 to 0 to +7
Binary = 2
1 0 0 0