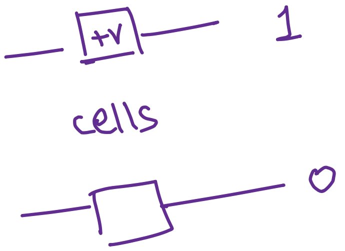
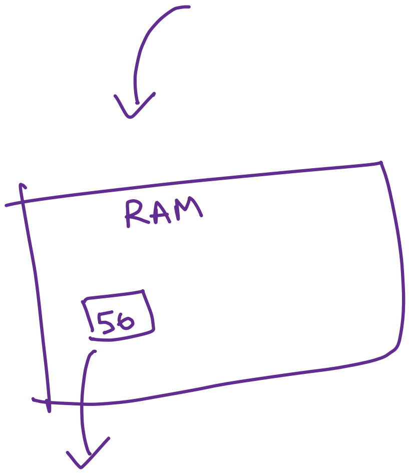
Electronic Circuit





Binary 0s and 1s

Decimal Number   Base 10

0, 1, 2, 3, 4, 5, 6, 7, 8, 9

$$\begin{array}{ccc} 10^2 & 10^1 & 10^0 \\ \hline 2 & 4 & 3 \end{array}$$

$$\hookrightarrow 2 \times 10^2 + 4 \times 10^1 + 3 \times 10^0 = 200 + 40 + 3$$

$$= \boxed{243}$$

Binary Number   Base 2

0, 1

$$\begin{array}{cccc} 2^3 & 2^2 & 2^1 & 2^0 \\ \hline 1 & 1 & 0 & 1 \end{array} = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 8 + 4 + 0 + 1$$

$$= 13$$

"Pen & Paper"

2 | 19
2 | 9, 1
2 | 4, 1
2 | 2, 0
2 | 1, 0
 | 0, 1

Top

Bottom

Base 2

$\boxed{19}$

2×9 = 18, 1

$2^4\ 2^3\ 2^2\ 2^1\ 2^0$

$\boxed{1\ 0\ 0\ 1\ 1}$ ✓ correct

$= 2^4 + 0 + 0 + 2^1 + 1$

$= 16 + 2 + 1$

$= \boxed{19}$

Schedule

Tue, Thu, Sat

Next week

9 – 11 pm

Divison Method

2 | 33
2 | 16, 1
2 | 8, 0
2 | 4, 0
2 | 2, 0
2 | 1, 0
 | 0, 1

$\{2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

$\boxed{1\ 0\ 0\ 0\ 0\ 1}$

$32 + 1 = \boxed{33}$

Intutive way

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

128 64 32 16 8 4 2 1

1

1 ~~23~~

1110001

33

113

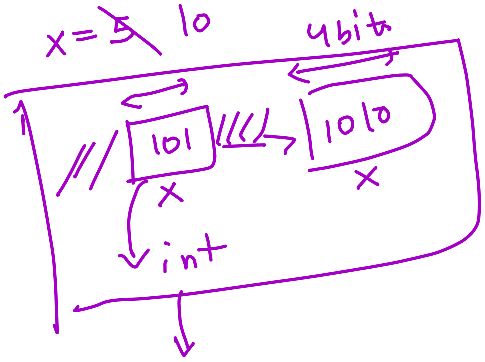
Negative Numbers || Storage ?

	32 bit
1	01
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010
288	✓

N inc

$x = 2 \times X$

More bits for a larger [✓ no]



fixed size

4bytes = 32bits

Puts a limit on the largest

2 bits

1 1 $= 3 = 2^2 - 1$

3 bits

1 1 1 $= 7 = 2^3 - 1$

1 1 1 1 $= 15 = 2^4 - 1$

11111111111111 $= 2^{32} - 1 \approx 10^9$

← 32 bits →

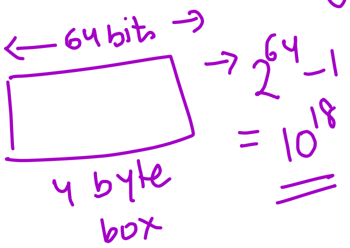
int

C++

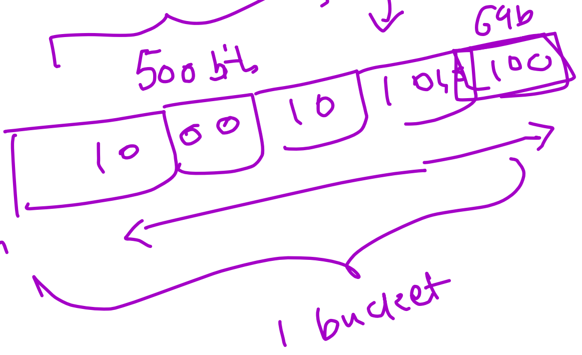
Java

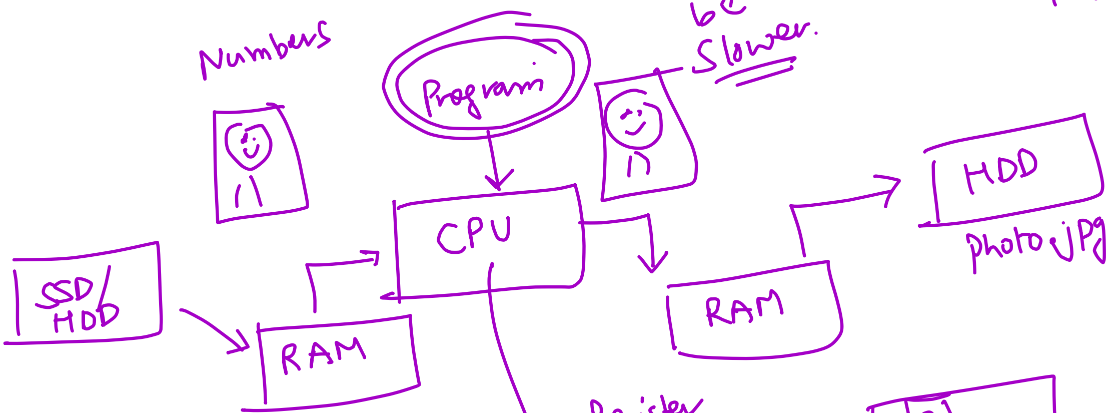
long → 64 bits

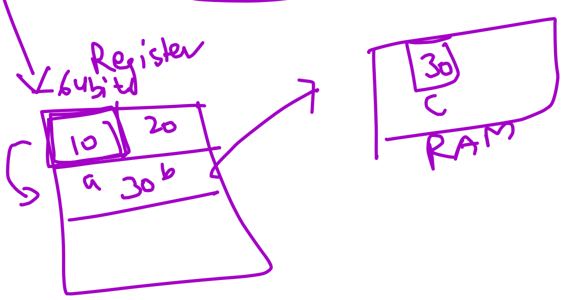
Python : automatically manages Store ~~for~~ you.



Addition in parts operation would be slower.







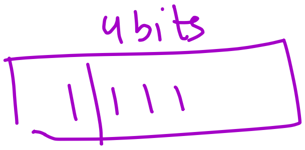
a = 10
b = 20
c = a + b

⇒ Memory

Negative Numbers

Python
Java

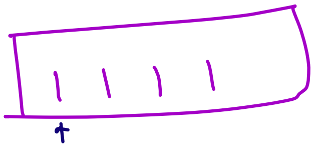
Signed



MSB denotes the sign

1 → -ve

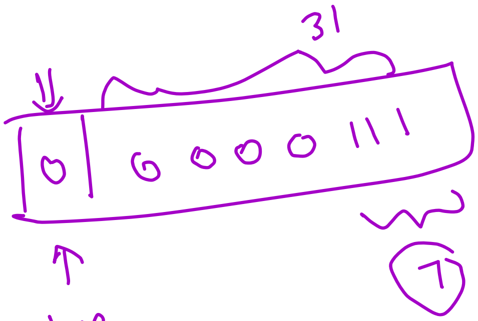
0 → +ve



No concept of sign bit.

" unsigned "

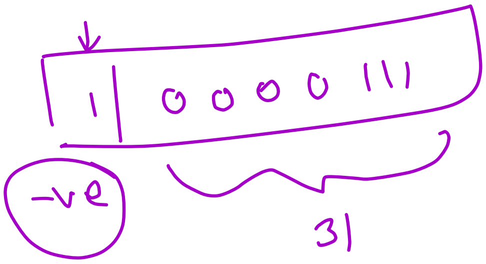
(~~unsigned int x = 10;~~)



+ve

→ $2^{32}-1$

$= 2^{31}-1$



$= -2^{31}$

2s Compliment Form

→ way of storing -ve numbers

→ why?

Adder ↔ Electronic circuit → | 10 | 20 |

10 + 20 = 30

10 - 20 = -10

subtracter X ~~Electronic Circuit~~

cost & complexity

100 Additions ✓

10 subtraction ✓

↓

1000 subtraction ✓

10 Additions ✓

Simplify the hardware

2s compliment of NO → -ve

5 - 5 = 0

$5 + (-5) = 0$

Addition

$$\begin{array}{lcccc}
\Rightarrow & 0 & 1 & 0 & 1 \\
\text{Flip all bits} & 1 & 0 & 1 & 0 \\
+1 & 0 & 0 & 0 & 1 \\
\hline
 & 1 & 0 & 1 & 1
\end{array}$$

0 1 0 1 ⇒ +5

1 0 1 1 → Represent -5

Addition

0 + 0 = 0

1 + 0 = 1

0 + 1 = 1

1 + 1 = 10

$$\begin{array}{rrcccc}
 & 5 & 0 & 1 & 0 & 1 \\
+ & -5 & 1 & 0 & 1 & 1 \\
\hline
 & & 0 & 0 & 0 & 0
\end{array}$$

Discarded

4bit

2^1 2^0

$\boxed{0}$

$\boxed{3}$ 0 0 1 1

-3

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

32 bit

$\boxed{0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1}$

32 bit

$\boxed{0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1}$

4 bit box

-3 +5

$$\begin{array}{r} +\ 0\ 1\ 0\ 1 \\ \hline \boxed{0\ 0\ 1\ 0} \end{array} \Rightarrow \boxed{2}$$

~~+5~~

$\boxed{1}$ x

← 4 bit →

2s compliment

$\boxed{-3}$

$\boxed{1 | 1\ 0\ 1}$ → Read what NO

↑ -ve no

$\underbrace{1\ 0\ 1}_{x}$

$-(-x) = +x$

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

⇓

$\boxed{3}$

$2 \times 2 \times 2 \times 2 = 16$ combinations

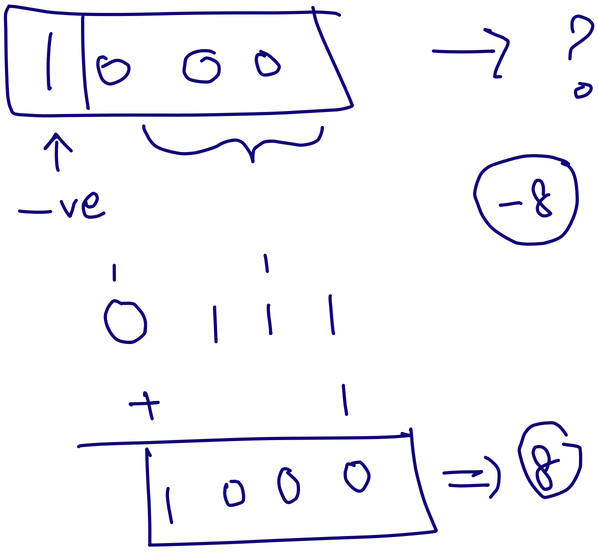
4 bits

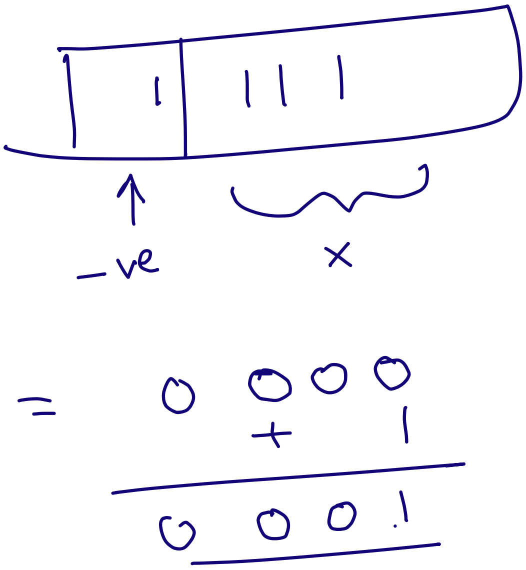
$\boxed{0 | 1\ 1\ 1} \Rightarrow +7$

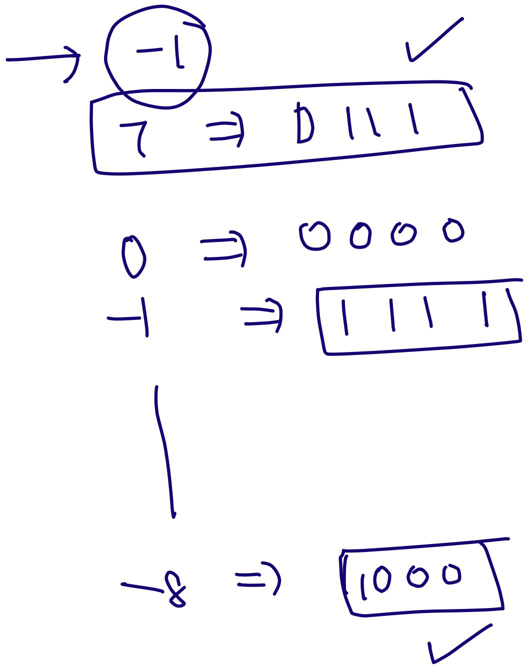
$\boxed{1 | 0\ 0\ 0} \Rightarrow -8$

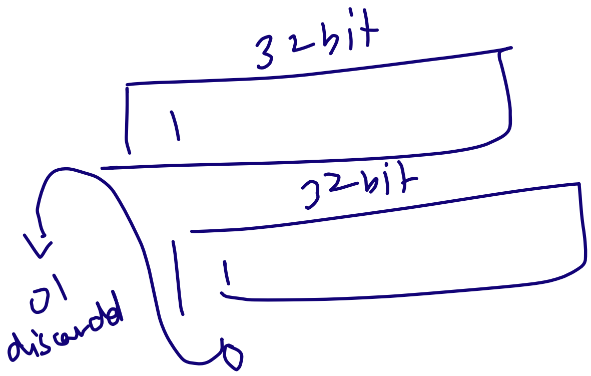
16 No's

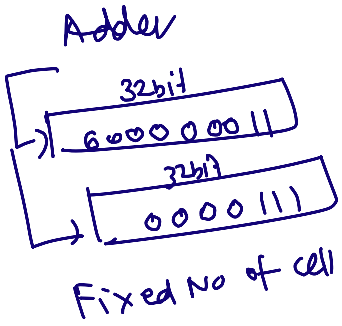
$\underbrace{-8 \_\_\_\_ -1,}_{8 \text{ No's}} \underbrace{0, \_\_\_\_ 7}_{8 \text{ No's}}$


1 0 0 0
→ ?
-ve
-8
0 1 1 1
+ 1
1 0 0 0 ⇒ 8


1 1 1 1
-ve
x
= 0 0 0 0
+ 1
0 0 0 1


→ -1
7 ⇒ 0111
0 ⇒ 0000
-1 ⇒ 1111
-8 ⇒ 1000


32bit
1
32bit
1
0
01
discarded


Adder
32bit
00000011
32bit
0000111
Fixed No of cell

signed

$$\left.\begin{array}{l} x = 5 \ ✓ \\ x = -5 \ ✓ \end{array}\right\}$$

x [illegible]

=

2s complim $\frac{1}{\quad}$

+ -

$$-2^{31} + 2^{31} - 1$$