Todays Content

✓ 1. Number System Basic
✓ 2. Binary to Decimal & ViceVersa
✓ 3. Adding 2 Binary Numbers
4. -ve numbers
5. Datatype range
6. When we multiply 2 numbers

$$\boxed{S = 2^0 + 2^1 + 2^2 + \cdots + 2^{N-2} + 2^{N-1} + 2^N = 2^{N+1} - 1}$$

$$2S = \cancel{2^1} + \cancel{2^2} + \cancel{2^3} + \cdots + \cancel{2^{N-1}} + \cancel{2^N} + 2^{N+1}$$

$$-S = 2^0 + \cancel{2^1} + \cancel{2^2} + \cdots + \cancel{2^{N-2}} + \cancel{2^{N-1}} + \cancel{2^N}$$

$$S = 2^{N+1} - 1$$

Ex1: $S = 2^0 + 2^1 + 2^2 = 2^3 - 1 = 7$

Ex2: $S = 2^0 + 2^1 + 2^2 + 2^3 = 2^4 - 1 = 15$

Decimal Number System

→ Each Digit : [0 1 2 .. 9]
→ power : [10]

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

$3\;4\;2 = 3*10^2 + 4*10^1 + 2*10^0 = 342$

$2\;5\;6\;3 = 2*10^3 + 5*10^2 + 6*10^1 + 3*10^0 = 2563$

Binary Number System

→ Each Digit : [0 1]
→ Each power : [2]

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

$= 2^5*1 + 2^2*1 + 2^0*1 = 32+4+1 = 37$

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

$= 2^4*1 + 2^1*1 + 2^0*1 = 16+2+1 = 19$

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

$= 2^4 + 2^3 + 2^0 = 16+8+1 = 25$

2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	1	0	1	0

$= 2^6 + 2^4 + 2^3 + 2^1 = 64+16+8+2 = 90$

Decimal to Binary

: # divide number/2 till it becomes = 0

1. At each step write remainders
2. All remainders bottom to up.

2	37	
2	18	: 1
2	9	: 0
2	4	: 1
2	2	: 0
2	1	: 0
	0	: 1

2	45	
2	22	: 1
2	11	: 0
2	5	: 1
2	2	: 1
2	1	: 0
	0	: 1

	2^5	2^4	2^3	2^2	2^1	2^0
37 :	1	0	0	1	0	1

	2^5	2^4	2^3	2^2	2^1	2^0
45 =	1	0	1	1	0	1

Add 2 Decimal numbers : d = sum%10   c = sum/10 } 10: Decimal system

17/10 16/10

		1	1	
11/10		7	8	9
	1	3	8	7
s =		11%10	17%10	16%10
d =	1	1	7	6

Add 2 Binary Numbers d = sum%2   c = sum/2 } 2: Binary system

	2^4	2^3	2^2	2^1	2^0	
	1/2	3/2	2/2	1/2		Decimal
c =	0	1	1	0		
1/2	1	0	1	1	0	→ 22
0	0	0	1	1	1	+ 7
s =	1%2	1%2	3%2	2%2	1%2	
d = 0	1	1	1	0	1	29

	2^5	2^4	2^3	2^2	2^1	2^0	
	3/2	2/2	2/2	3/2	2/2		
c =	1	1	1	1	1		
1/2	0	1	1	0	1	1	→ 27
0	0	1	0	1	1	1	→ 23
s =	1%2	3%2	2%2	2%2	3%2	2%2	
d = 0	1	1	0	0	1	0	→ 50

8 bit Numbers:

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
13 :	0	0	0	0	1	1	0	1
-13 :	1	0	0	0	1	1	0	1

* Wrong way ?

This way of representing -ve numbers is wrong

carry				1	1	0	0	
-13 :	1	0	0	0	1	1	0	1
4 :	0	0	0	0	0	1	0	0
			2^5	2^4	2^3	2^2	2^1	2^0
Ideally = -9 ≠ -17 :	1	0	0	1	0	0	0	1

$= 2^4 + 2^0 : 17$

Q? Correct way to get negative:

$-a = 2's\ a : \sim a\ [1 \rightleftarrows 0] + 1$

a = 13 :	0	0	0	0	1	1	0	1
	1/2	1/2	1/2			1/2	1/2	
carry	0	0	0	0	0	0	0	
$\sim a$:	1	1	1	1	0	0	1	0
1 :	0	0	0	0	0	0	0	1
	1%2	1%2	1%2	1%2	0	0	1%2	1%2
$-a$:	$\boxed{1}$	1	1	1	0	0	1	1
	$\boxed{2^7}$	2^6	2^5	2^4	2^3	2^2	2^1	2^0

$= 2^7 + 2^6 + 2^5 + 2^4 + 2^1 + 2^0 = 128 + 64 + 32 + 16 + 2 + 1 = 243 \neq -13.$

LMB : MSB = most significant Bit & base value of it is -ve.

$2^7 > \{2^6 + 2^5 + 2^4 + \cdots 2^0\}$

$128 > 127$

$-a$:	1	1	1	1	0	0	1	1
	-2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

$= -2^7 + 2^6 + 2^5 + 2^4 + 2^1 : -128 + 64 + 32 + 16 + 2 + 1 : -128 + 115 = -13$

Q: Will base value of MSB bit be always -ve : No

[unsigned int a;
unsigned long b;]

#bits	Signed numbers: MSB value -ve (default)	Unsigned numbers: MSB value +ve
4	$\underset{-2^3}{0}\ \underset{2^2}{1}\ \underset{2^1}{1}\ \underset{2^0}{1}$ $\overset{-2^3}{1}\ \overset{2^2}{0}\ \overset{2^1}{0}\ \overset{2^0}{0} = -2^3$ $\overset{-2^3}{0}\ \overset{2^2}{1}\ \overset{2^1}{1}\ \overset{2^0}{1} = 2^3-1$	$\underset{2^3}{1}\ \underset{2^2}{1}\ \underset{2^1}{1}\ \underset{2^0}{1}$ $\overset{2^3}{0}\ \overset{2^2}{0}\ \overset{2^1}{0}\ \overset{2^0}{0} = 2^0$ $\overset{2^3}{1}\ \overset{2^2}{1}\ \overset{2^1}{1}\ \overset{2^0}{1} = 2^4-1$
8	$\underset{-2^7}{0}\ \underset{2^6}{1}\ \underset{2^5}{1} \dots \underset{2^1}{1}\ \underset{2^0}{1}$ $\overset{-2^7}{1}\ \overset{2^6}{0}\ \overset{2^5}{0} \dots \overset{2^1}{0}\ \overset{2^0}{0} = -2^7$ $\overset{-2^7}{0}\ \overset{2^6}{1}\ \overset{2^5}{1} \dots \overset{2^1}{1}\ \overset{2^0}{1} = 2^7-1$	$\underset{2^7}{1}\ \underset{2^6}{1}\ \underset{2^5}{1} \dots \underset{2^1}{1}\ \underset{2^0}{1}$ $\overset{2^7}{0}\ \overset{2^6}{0}\ \overset{2^5}{0} \dots \overset{2^1}{0}\ \overset{2^0}{0} = 0$ $\overset{2^7}{1}\ \overset{2^6}{1}\ \overset{2^5}{1} \dots \overset{2^1}{1}\ \overset{2^0}{1} = 2^8-1$
N	$\underset{-2^{N-1}}{1}\ \underset{2^{N-2}}{0}\ \underset{2^{N-3}}{0} \dots \underset{2^1}{0}\ \underset{2^0}{0}$ $\overset{-2^{N-1}}{1}\ \overset{2^{N-2}}{0}\ \overset{2^{N-3}}{0} \dots \overset{2^1}{0}\ \overset{2^0}{0} = -2^{N-1}$ $\overset{-2^{N-1}}{0}\ \overset{2^{N-2}}{1}\ \overset{2^{N-3}}{1} \dots \overset{2^1}{1}\ \overset{2^0}{1} = 2^{N-1}-1$	$\underset{2^{N-1}}{0}\ \underset{2^{N-2}}{0}\ \underset{2^{N-3}}{0} \dots \underset{2^1}{0}\ \underset{2^0}{0}$ $\overset{2^{N-1}}{0}\ \overset{2^{N-2}}{0}\ \overset{2^{N-3}}{0} \dots \overset{2^1}{0}\ \overset{2^0}{0} = 0$ $\overset{2^{N-1}}{1}\ \overset{2^{N-2}}{1}\ \overset{2^{N-3}}{1} \dots \overset{2^1}{1}\ \overset{2^0}{1} = 2^N-1$

Datatype Ranges:

#datatype	#bits N	Signed: Min -2^{N-1}	Max $2^{N-1}-1$	Unsigned: Min 0	Max 2^N-1
byte	8	Signed: -2^7 .. 2^7-1 $[-128 .. 127]$		Unsigned: $[0 ... 2^8-1]$ $[0 .. 255]$	
int	32	Signed: $[-2^{31} .. 2^{31}-1]$ $[-2*10^9 .. 2*10^9]$		Unsigned: $[0 .. 2^{32}-1]$ $[0 .. 4*10^9]$	
long	64	Signed: $[-2^{63} .. 2^{63}-1]$ $[-8*10^{18} .. 8*10^{18}]$		Unsigned: $[0 ... 2^{64}-1]$ $[0 .. 16*10^{18}]$	

Approximations: $2^{10} = 1024 \approx 1000 = 10^3 \Rightarrow \boxed{2^{10} \approx 10^3}$

Case 1: Calculate 2^{31}

$2^{10} \approx 10^3$

Apply cube on both sides

$(2^{10})^3 \approx (10^3)^3$

$2^{30} \approx 10^9$

Multiply 2 on both sides

$\boxed{2^{31} \approx 2*10^9}$

Multiply 2 on both sides

$\boxed{2^{32} \approx 4*10^9}$

Case 2: Calculate 2^{63}

$2^{10} \approx 10^3$

Apply power 6 on both sides

$(2^{10})^6 \approx (10^3)^6$

$2^{60} \approx 10^{18}$

Multiply 8 on both sides

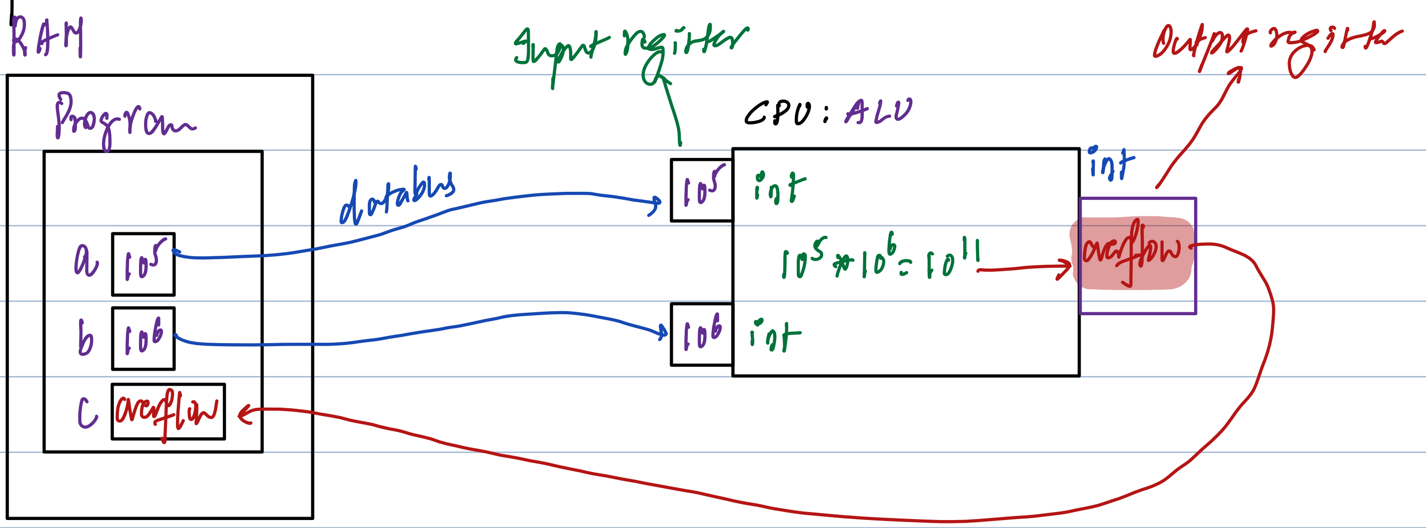
$\boxed{2^{63} \approx 8*10^{18}}$

Multiply 2 on both sides

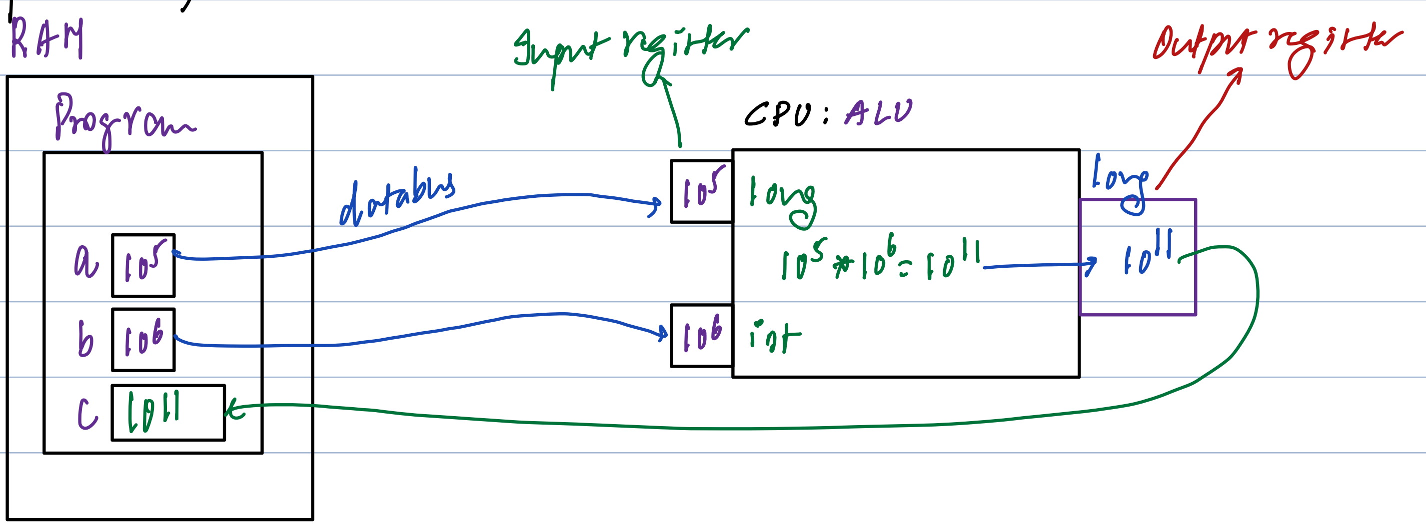
$\boxed{2^{64} \approx 16*10^{18}}$

Q1: int a = 10^5, b = 10^6;
int c = a*b; // $10^5 * 10^6 = 10^{11}$ > int range *
print(c); * Wont get 10^{11}

Q2: ✓int a = 10^5, b = 10^6;
✓long c = a*b;
print(c);



Q3: int a = 10^5, b = 10^6;
long c = (long) a * b;
print(c);



#Note: When we multiply 2 int, result will be int type & it can overflow, which can give you issues in code. Carefull Debugging