Doubt

#

```
      1
    2 3 2
  3 4 5 4 3
4 5 6 7 6 5 4
```

# Star Pattern

Observations

N = 3

```
_ _ * \n
_ * * * \n
* * * * * \n
```

1. Rows is equal to N
2. N-i Spaces, 2i-1 Stars in row, New Line \n

observation

		Spaces	= N-i	Stars	= $\boxed{2i-1}$
i=1	_ _ *	2	3-1 = 2	1	= 2(1)-1 = 1
i=2	_ * * *	1	3-2 = 1	3	2(2)-1 = 3
i=3	* * * * *	0	3-3 = 0	5	2(3)-1 = 5

- Alternative

$\overset{d}{\longleftrightarrow}\overset{d}{\longleftrightarrow}$

$1, 3, 5, \dots n^{th}$

$\uparrow$ a

$d = 2$

"Airthmetic Progression"

a = first term

d = common diff

Proof $\Rightarrow$

$$\boxed{T_i = a + (i-1)d}$$

ith term

$$= 1 + (i-1)2$$

$$= 1 + 2i - 2$$

$$= \boxed{2i-1}$$

General term

Maths

$i = 8$

Series $\longleftrightarrow$ 3, 7, 11, 15 ..... $\boxed{8^{th}}$ Term

ith

$$T_i = a + (i-1)d$$

$$= 3 + (8-1)4$$

$$= 3 + 28$$

$$= \boxed{31}$$

$$3 + (i-1)4$$

$$= 3 + 4i - 4$$

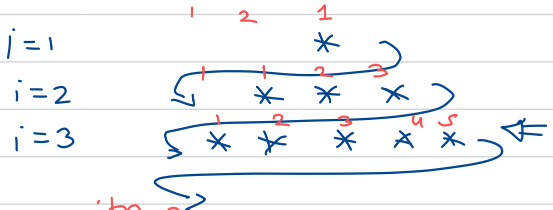
$$= \cancel{4i+2}\ 4i - 1$$

Pseudo code

Rows

```
Read N

i = 1
while i <= N {
    // spaces   N - i
    cnt = 1
    while cnt <= N-i {
        print " "
        cnt = cnt + 1
    }
    // stars   2i - 1
    cnt = 1
    while cnt <= 2i-1 {
        print "*"
        cnt = cnt + 1
    }
    print ("\n")
```



→ ith Row spaces

For ↔ while

2(1) − 1 = 1 star

2(2) − 1 = 3 stars

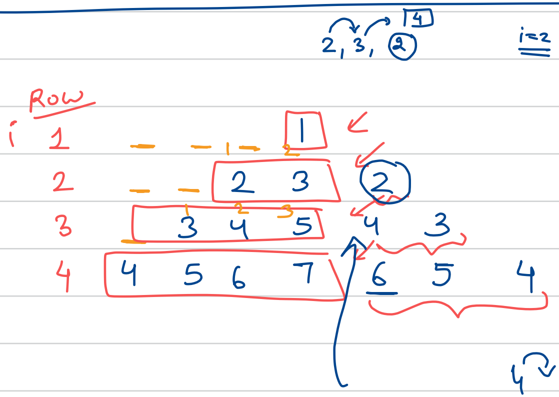
2(3) − 1 = 5 Stars

i = i + 1

}

exit

#Q



N=4

spaces	i inc	i−1 dec
4−1 = 3	1	0
4−2 = 2	2	1
4−~~B~~3 = 1	3	2
4−4 = 0	4	3

$\boxed{8}$ −2

4 → 5 → 6 → 7 6 → 5 → 4

Obs.

⇒ Rows equal to N

⇒ N−i spaces, i ↑ inc, i−1 ↓ dec

Starting from val i, starting → ? val−2

```
Read N
i = 1

while i <= N {
    // spaces
    cnt = 1
    while cnt <= n-i {
        print " "
        cnt = cnt + 1
    }

    // inc
    cnt=1 , val = i
    while cnt <= i {
        print val
        val = val + 1
        cnt = cnt + 1
    }

    // dec
```

3 4 5

i=3

cnt | ~~2~~ 3 4 val | ~~3~~ 4 5 6

1<=3

2<=3

3<=3

4<=3

6 → 4

i=3

| ~~5~~ 4 | ~~3~~ val

3 4 5 4 3

1 2

```
⇒ val = val - 2                      // Reset the
                                        Correct value
                                        for dec
                                        part.
  cnt = 1

  while cnt <= i-1 {

      ⇒ print (val)
      ⇒ val = val - 1
      ⇒ cnt = cnt + 1

  }

  // Next Row

  print ("\n")
  i = i + 1
}

exit
```

1<= 2 i=3

2<=2

~~3<=2~~

[Number Systems]

Computer Science

↓

Binary System [Base 2]

$\underbrace{0,1}$

Human → Decimal [Base 10]

$$\underbrace{0,1,2,3,4,5,6,7,8,9}_{\text{10 digit}}$$

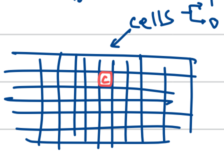
1248

$\boxed{321}$

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

$$= 3 \times 10^2 + 2 \times 10^1 + 1 \times 10^0 = 300 + 20 + 1 = \underline{\underline{321}}$$

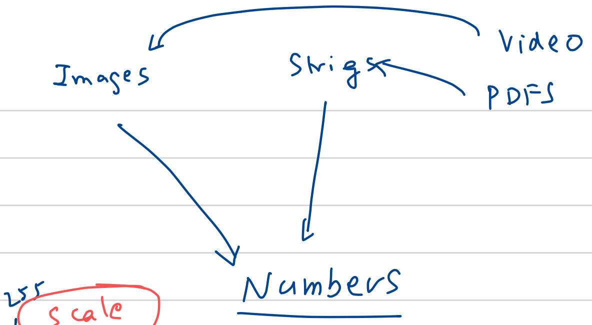
because

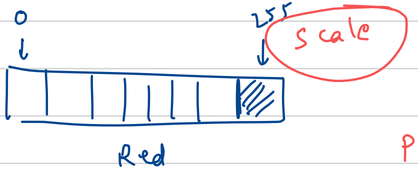


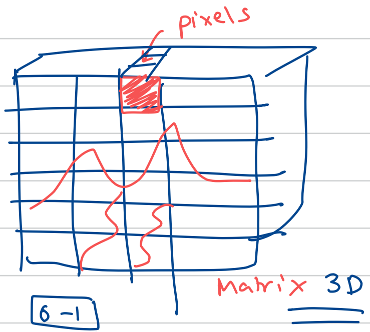
⇒ $\underbrace{\text{images, audio, files, pdf}}$ ---- ▷ Binary Representation?

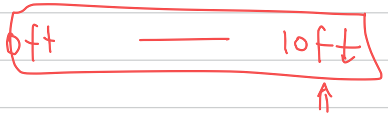
RAM ⇒ volalite

Disk ⇒ permanent









$$\text{Pixel} = [\overset{0-255}{R} + \overset{0-255}{G} + \overset{0-255}{B}]\ \overset{\boxed{0-1}}{\text{opacity}}$$

= (220, 10, 0) ⤳ Reddish

= (255, 0, 0) ⤳ Deep Red

= (255, 255, 255) ⤳ white

= (0, 0, 0) ⤳ Black

= (100, 100, 100) ⤳ grey

= (200, 200, 200) ⤳ light grey

How tne numbers are stored?

⇒ Binary

21

Base ⇒ 2

8	4	2	1
2^3	2^2	2^1	2^0
1	1	0	1

13

$= 2^3 \times 1 = 8$

$+ 2^2 \times 1 \quad +4$

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

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

$\underline{\quad\quad}$

13

3 2 1

$= 300 + 20 + 1$

16	8	4	2	1
1	1	0	1	1

$16 + 8 + 0 + 2 + 1$

$= \boxed{27}$

32	16	8	4	2	1
1	0	1	1	1	0

32 + 8 + 4 + 2

= 46

Decimal into Binary

2	46	
2	23 ,	0
2	11 ,	1
2	5 ,	1
2	2 ,	1
2	1 ,	0
	0 ,	1

1 0 1 1 1 0

N

2	17	
2	8 ,	1
2	4 ,	0
2	2 ,	0
2	1 ,	0
	0 ,	1

$17 = 2 \times \underline{8} + \underline{1}$

16	8	4	2	1
1	0	0	0	1

$= 16 + 1 = 17$

2	50	
2	25 ,	0
2	12 ,	1
2	6 ,	0
2	3 ,	0
2	1 ,	1
	0 ,	1

$2 \times 25 + 0$

$2 \times 12 + 1$

$2 \times 6 + 0$

$3 = 2 \times \underline{1} + \underline{1}$

$1 = 2 \times \underline{0} + \underline{1}$

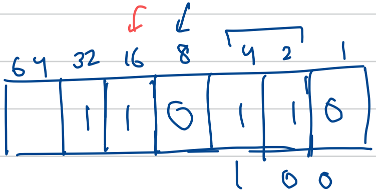
32	16	8	4	2	1
1	1	0	0	1	0

$32 + 16 + 2$

$= 48 + 2$

$= 50$

Shortcut



10110 → 22

100111 → 39

39 = 32 + 4 + 2 + 1

54 = 32 + 16

110110 → 54

1 → 1

2 → 10

3 → 11

4 → 100

5 → 101

6 → 110

7 → 111

8 → 1000

9 → 1001

10 → 1010

11 → 1011

image

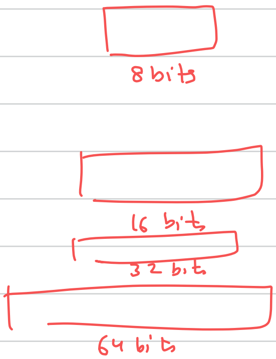
↓

pixels

↓

0-255

Store an int



Resolution

$\boxed{1\ 1} = 3$

2 bit

$\boxed{1\,1\,1} = 7$

3 bits

$\boxed{} = 15$

4 bits

⇒ $\boxed{11111111} = \boxed{255}$

8 bits

↑

8 bits

$\boxed{00000000\ 1111111}$ ⇒

64 bits

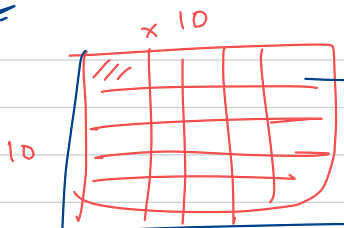
↑

DLSR

⇒ 6000x4000

⇒ 1920 x 10180

⇒ 1024 x 720



Pixel = (R, G, B)

↓

8 bit

$\boxed{1\text{ GB}} = \boxed{10 \times 10} \times 3 \times \boxed{8} = (x \text{ MB})$

1 MB

$8\text{GB} = \boxed{10 \times 10} \times 3 \times 64 = (8 \times \text{MB})$

8 MB

Smart phone

:(

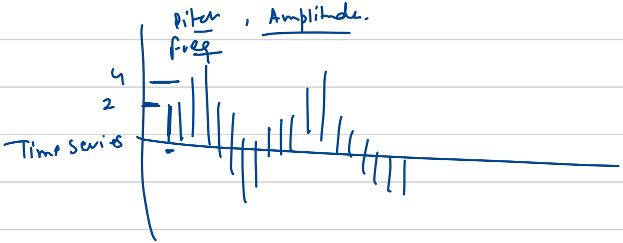
1000 image

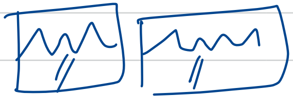
Pixel = (11, 7, 2)

= (1011, 111, 10)

= $\boxed{1011}\boxed{111}\boxed{10}$

24 bit





images + audio = video

Range

largest

$\boxed{1\ 1}$ = 3 2^2-1

2 bit

Short $\boxed{1\ 1\ 1\ 1}$ = 15 2^4-1 8 + 4 + 2 + 1

16 bit = $2^{16}-1$

⋮

int $\boxed{\qquad 5}$ = $2^{32}-1$ ($\simeq 10^9$) 2,147,483,647

32 bits

long $\boxed{\qquad 5}$ = $2^{64}-1$ ($\simeq 10^{18}$)

64 bits

4 bit

$\boxed{0\ 0\ 0\ 0}$ → 0

4 bit

⋮

$\boxed{1\ 1\ 1\ 1}$ → 15

0 to $2^4 - 1$

$= \boxed{0 - 15}$

sign magnitude

$\boxed{0 \mid 0\,0\,0}$ sign → 0 +ve

$\boxed{1 \mid 1\,1\,1}$ sign → 1 -ve

$\boxed{0 \mid 1\,1\,1}$ $\textcircled{7}$

4 bit

$\boxed{0 \mid \quad\quad}$

32 bit

0 to $\cancel{2^{32} - 1}$ $2^{31} - 1$

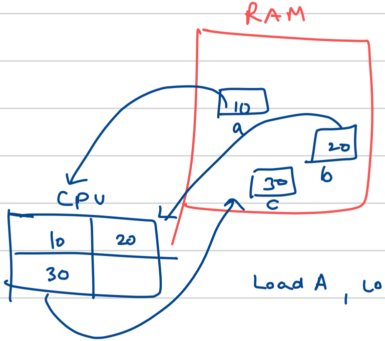
Processing by CPU

$a = 10$

$b = 20$

Adder

$C = \underbrace{a + b}$



32 bit / 64 bit

Load A , Load B

$C = \underline{\underline{b - a}}$

Additional circuit

Additional Hardware
Sub

80 Additions , ~~20 Subtractions~~

More Load ~~Less Load~~

Subtracter

$$5 \;\; - 5 \;\; = 0$$

$$5 + \underbrace{(-5)}_{\uparrow} = 0$$

2's compliment form

-ve are stored inside in 2's compliment form

-5

① Flip all bits of 5

② Add 1 to it

$\boxed{-5}$ $\quad \boxed{\boxed{0}\ 1\ 0\ 1}$

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

$\textcircled{-5} \longrightarrow \boxed{\boxed{1}\ 0\ 1\ 1}$ Memory

$$\begin{array}{rr} & 1\ 1\ \ 1\ 1 \\ 5 & 0\ 1\ 0\ 1 \\ -5 & 1\ 0\ 1\ 1 \\ \cancel{1} & 0\ 0\ 0\ 0 \end{array} \Rightarrow \textcircled{0}$$

← 4bit →

Binary Addition

$1+1 = 10$

$1+0 = 1$

$0+1 = 1$

$0+0 = 0$

0000 0101

1111 1010

−5 1111 1011

+5 0000 0101

1 0000 0000

← 8 bit →

11 0000 1011

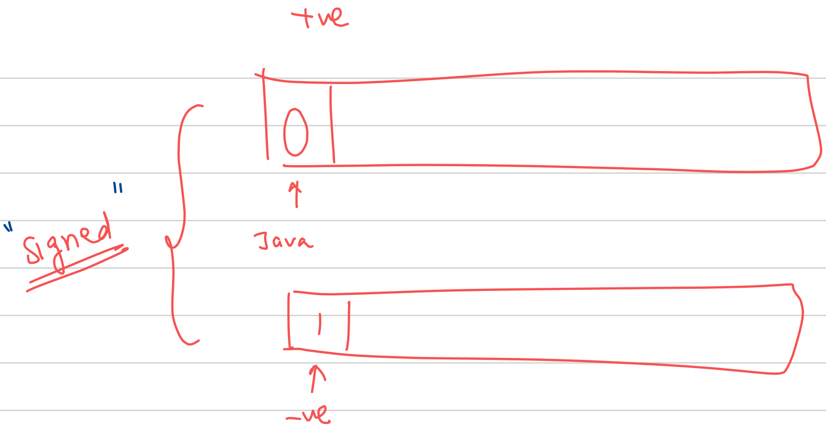
−5 1 1111 1011

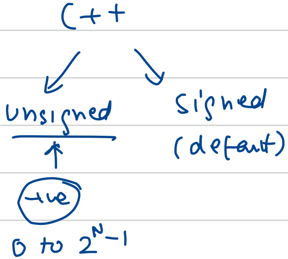
0000 0110

+ve 4 2 1

= 6

11 − 5 = 6

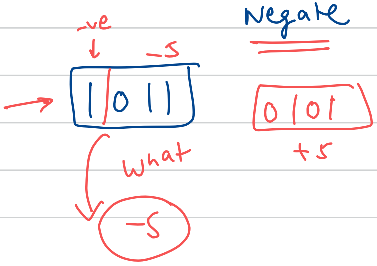




~~X 0 0 0 0 1~~ → -1 No

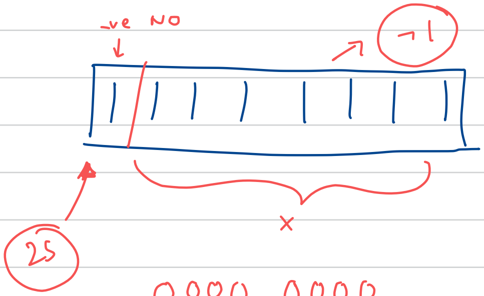
Binary of a -ve No ↷ 2s compliment form

$-(-5) \Rightarrow +5$



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

$4+1 = 5$



$-(-x)$

$= +x$

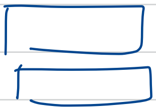
```
  0000 0000
+         1
-----------
  0000 0001
```

```
  1111 1111
+ 0000 0001
-----------
  0000 0000
```

32 bit

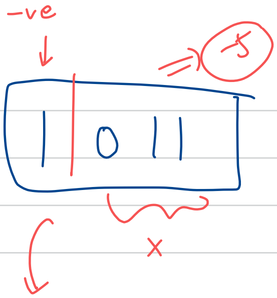
$15 + 15 = \boxed{30}$ 5 bits

4 bit



```
        1  1 1
= 0000 1111
= 0000 1111
-----------
  0001 1110
-----------
     16 8 4 2 1
```

$16 + 8 + 4 + 2 = \boxed{30}$



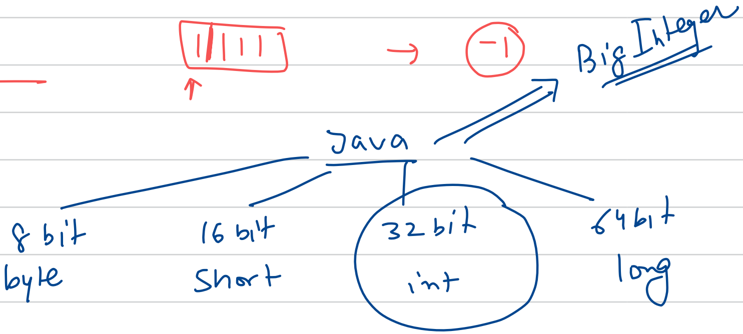
$-(-x)$

$= \boxed{+x} \Rightarrow$ Human

```
  0 1 0 0
+ 0 0 0 1
---------
  0 1 0 1
```

0 | 0 0 0 1 1 1 1 → 15

1 | 1 1 1 1 → $\boxed{-1}$



50 → 32bit 0 0 0 0 0 0 1 1 0 0 1 0

Pseudo code ✓

P

HW → ① Decimal to Binary

HW → ② Binary to Decimal

Loop

Extract the Digits

24

$$\begin{array}{ccccc} 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\ 1 & 0 & 1 & 1 & 0 \end{array}$$

$N = \boxed{1\ 0\ 1\ 1\ 0}$ ↑

0×2^0

$+ 1 \times 2^1$

$+ 1 \times 2^2$

$+ 0 \times 2^3$

$+ 1 \times 2^4$

Q Sum of Digits of N.

$N = \cancel{3}\ \cancel{1}\ \cancel{2}\ \cancel{8}$

$\frac{3}{10} = 0$

$8+1+2+3$

$= 14$

```
Read N
  Sum = 0
```

$N = 3218$

Sum

$\boxed{0 \; +8 \; +1}$ $+2$ $+3$

```
while ( N > 0 ) {
    → rem = N % 10
    → sum = sum + rem
    → N = N/10
}
print sum
    exit
```

rem = 3218 % 10 = 8   = 14

$\underline{321 \times 10} + \boxed{8}$

N = $\frac{3218}{10}$ = 321

rem = 1

~~7018 % 10~~ N = 32

N = 3

N = 0

ASCII

Maps every char to NO

@	–
"	–
,	–
$	–
%	–

Number

a	97 → 1011110
b	98
c	99
⋮	

A	65
B	66
⋮	

[MWF
TTS

Next week

[Java   9-11 PM
Prateek

Same

12 classe

Unicode

न म ₹

₹ → Number → Binary

Recommend

☆
- → Intellij idea (Community Edition)
- → javac (compiler)

((Online ide IB Ide / CM ide))