- Introduction to Problem Solving
- Time Complexity
- Introduction to Arrays
- Prefix Sum
- Carry Forward & Subarrays
- Sliding Window & Contribution Technique
- Memory Management
- Sorting Basics
- 2D Matrices
- Bit Manipulations Basics
- Strings
- Interview Problems
- Contest [covers Full Intermediate DSA]

Following will be covered today!

1. Count the Factors
2. Optimisation for counting the Factors
3. Check if a number is Prime
4. Sum of N Natural Numbers
5. How to find the number of a times a piece of code runs, i.e, number of Iterations.
6. How to compare two Algorithms.

Question: No. of factors

Factor of N: A number which divides N

32: 1, 2, 4, 8, 16, 32 => 6

24: 1, 2, 3, 4, 6, 8, 12, 24 => 8

10: 1, 2, 5, 10 => 4

How to check if i is a factor of N?

- $N \% i == 0$ (Remainder when N / i)

Approach 1: Brute Force (Naive Solution)

1st factor of N: 1

Last factor of N: N

```
int countFactors(int N){
    int count = 0;
    for(int i=1; i<=N; i++){
        if(N%i==0){
            count++;
        }
    }
    return count;
}
```

N iterations

#Iterations = N

1 sec => 10^8 iterations

$10^9 = 10^8 \times 10^1$

10^8 → 1 sec

$10^8 \times 10 = 10^9$ → 10 sec

N	Iterations	Time
10^8	10^8	1sec
10^9	10^9	10 sec
10^{10}	10^{10}	100 sec
⋮		
10^{18}	$10^8 \times 10^{10}$	10^{10} sec = 317 years

Approach2:

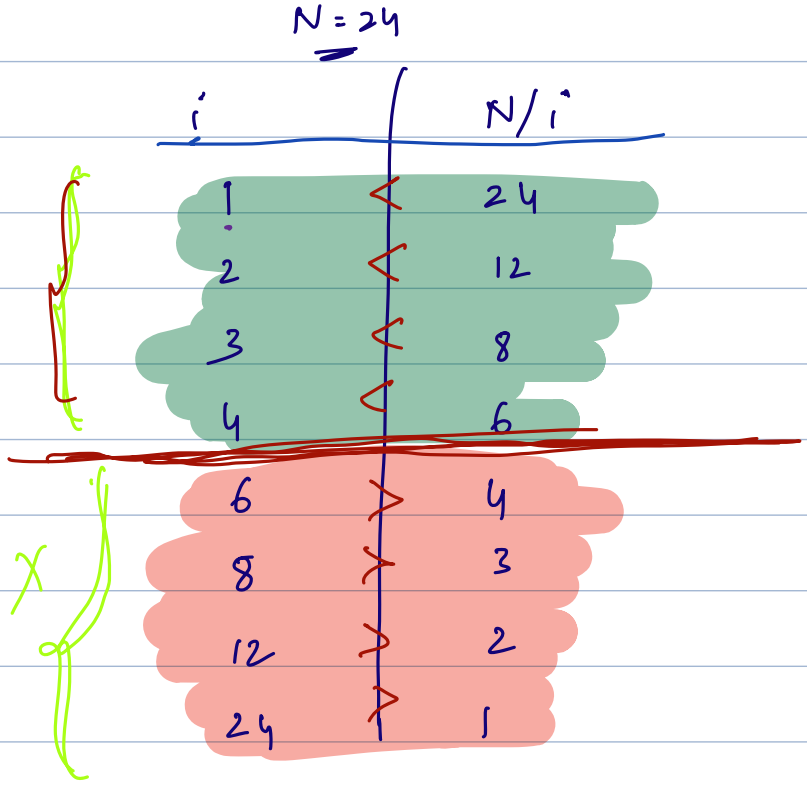
Observations:

3 is a factor of 24, do we know any other factor using this info?

$$\frac{24}{3} = 8 \Rightarrow \frac{24}{8} = 3$$

$\Rightarrow$ 4 is factor $\rightarrow$ $\frac{24}{4} = 6$ is also a factor

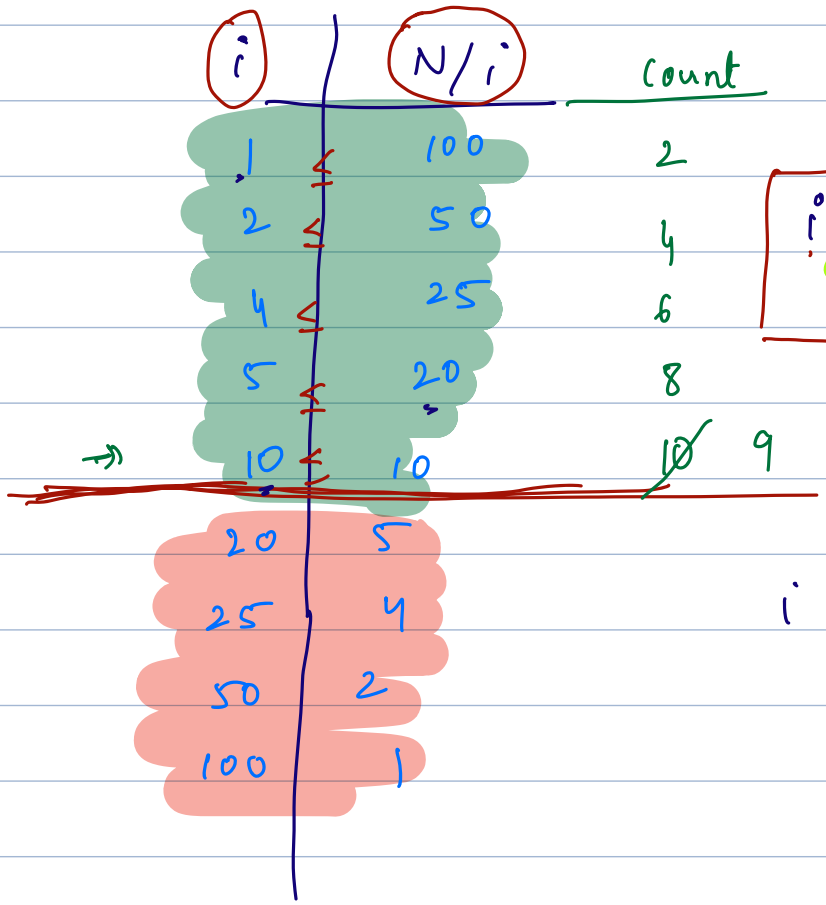
If i is a factor, $\frac{N}{i}$ is also a factor



Obs :- The factors are repeating after a certain point.

- We can only consider the first half to get all our factors

N = 100



$i \le \frac{N}{i} \Rightarrow i \times i \le N$

$i^2 \le N$

$i \le \sqrt{N}$

$i > \frac{N}{i}$

```
int factors(int N) {
    int count = 0;
    for(int i = 1; i*i <= N; i++) {        // i <= N/i
        if(N%i == 0) {
            if(i == N/i) {
                count = count + 1;
            }
            else {
                count = count + 2;
            }
        }
    }
    return count;
}
```

$\sqrt{N}$ iterations

iterations =

$i \times i \leq N \Rightarrow \sqrt{i^2} \leq \sqrt{N} \Rightarrow i \leq \sqrt{N}$

Range of i: $[1, \sqrt{N}]$

N	# iterations ($\sqrt{N}$)	Time
10^{18}	$\sqrt{10^{18}} = 10^9$	10 sec

Question: Prime Number

A number which has exactly 2 factors (1 and N)

2, 3, 5, 7, 11, 13, 17, 19, 23 .....

Quiz: 10 11 23 2 25 27 31

```
boolean isPrime(int N) {
    if(factors(N) == 2) {
        return true;
    }
    else
        return false;
}
```

#iterations = $\sqrt{N}$

$N = 10^{18}$ => 10^9 iters => 10sec

8:34am

Range

[a, b] => All numbers from a to b where a and b are also included

closed bracket

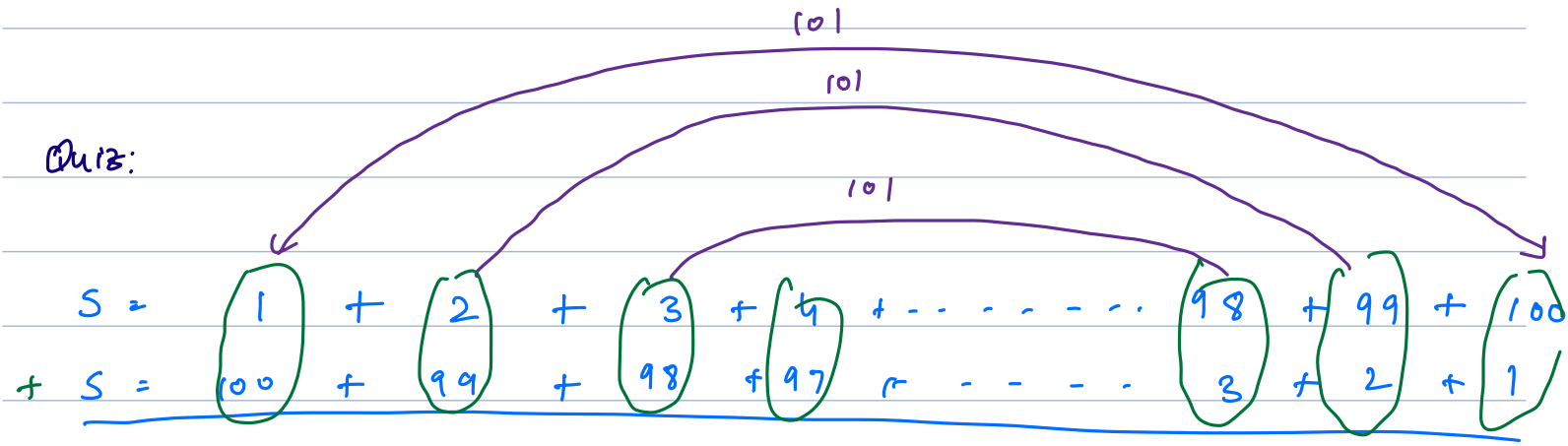
(a, b) => All numbers from a to b where a and b are not included

[3, 10] => 3, 4, 5, 6, 7, 8, 9, 10 => 8

(10 - 3 + 1)

[a, b] => b − a + 1

Quiz:



2S = 101 + 101 + 101 + 101 . . . . . 101 + 101 + 101

100 times

2S = 101 × 100

S = $\frac{101 \times 100}{2}$ = 101 × 50 = 5050

$$S = 1 + 2 + 3 + \cdots + N-2 + N-1 + N$$

$$\Rightarrow S = N + (N-1) + (N-2) + \cdots + 3 + 2 + 1$$

$$2S = \underbrace{(N+1) + (N+1) + (N+1) + \cdots + (N+1) + (N+1) + (N+1)}_{N \text{ times}}$$

$$2 \cdot S = N(N+1)$$

$$\boxed{S = \frac{N(N+1)}{2}}$$

Iterations: No. of times a loop runs

Quiz:

```
for (i = 1; i ≤ N; i++) {
    → if (i == N) {
          break;
      }
}
```

i = 1, 2, 3 ..... N

N times

Quiz:

How many iterations will be there in this loop ?

```
for(i -> 0 to 100){
    s = s + i + i^2;
}
```

Range of i : [0, 100]

[a, b] : b − a + 1

[0, 100] = 100 − 0 + 1 = 101 iteration

Quiz:

How many iterations will be there in this loop?

```
func(){
    for(i -> 1 to N){
        if(i % 2 == 0){
            print(i);
        }
    }
    for(j -> 1 to M){
        if(j % 2 == 0){
            print(j);
        }
    }
}
```

N

M

N + M

Geometric Progression

Series of numbers where the ratio of consecutive terms is always same

Ex: 3 6 12 24 48 96 192

$\frac{6}{2} = 2$ $\frac{12}{6}$ $\frac{24}{12}$ $\frac{48}{24}$ $\frac{96}{48}$ $\frac{192}{96}$

Ex: 4 12 36 72

$\frac{12}{4} = 3$ $\frac{36}{12} = 3$ $\frac{72}{36} = 2$

Ex: 5 (×2) 10 (×2) 20 (×2) 40 80 160 320 640 ...

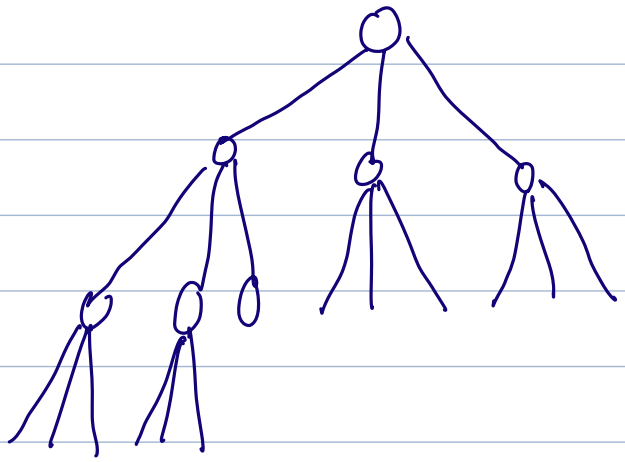
a: First term of GP

r: common ratio

Ex: a, ar, ar^2, ar^3, ar^4, ... ar^{K-1}

(1st, 2nd, 3rd, 4th, 5th, ..., Kth)

$$Sum_K = \frac{a(r^K - 1)}{r - 1}$$



$3^0 = 1$

3^1

3^2

3^3

3^4

$3^0 \quad 3^1 \quad 3^2 \quad 3^3 \quad 3^4 \quad \ldots \ldots \Rightarrow$ GP

How to compare 2 Algos ?

Divya (Div Sort)

15 sec (Windows xp)

↓ Macbook

7 sec (C++)

↓

7 sec

Teja (Sort Numbers)

10 sec (Macbook M4)

↓

10 sec (Python)

↓ C++

5 sec

Execution time is dependent on lots of factors on which we don't have control

#iterations is always constant

```
for (i = 1; i ≤ N; i++) {
    print("Hi");
}
```

x = 2 => i

x = a + b

```
For (i = 1; i <= N; i++){
    a = 10;
    b = 30;
    c = 40;
}
```

$= 10^{10}$

$N \times 3 : 3N$

$\Rightarrow \quad 10^{10} = 10^{8} \times 10^{2} = 100$ sec

(10^{8} sec)

$10^{18} = 10^{8} \times 10^{10} \Rightarrow 10^{10}$ sec

(10^{8} sec)

$10^{8} \rightarrow 1$ sec

$10^{18} \rightarrow ?$

$\frac{10^{18} \times 1 \text{ sec}}{10^{8}}$

$10^{8} \times 10^{8} \times 10^{2} = 100 \text{ (sec)}^2$

$100 \times 10^{8} \times 10^{8}$

10^{8} sec $\quad$ 10^{8} sec

Derivation of GP formula

$$(r^n - 1) = (r-1)(1 + r + r^2 + r^3 + \cdots + r^{n-1})$$

$$\boxed{\frac{r^n - 1}{r-1}} = 1 + r + r^2 + \cdots + r^{n-1}$$

$$a \quad ar \quad ar^2 \quad ar^3 \cdots \cdots \cdots ar^{K-1}$$

K^{th}

Sum of K terms: $a + ar + ar^2 + \cdots ar^{K-1}$

$$= a(1 + r + r^2 + \cdots r^{K-1})$$

$$= \boxed{\frac{a(r^K - 1)}{r - 1}}$$

$$r^2 - 1 = (r-1)(r+1)$$

$n = 2$

$$r^2 - 1 = (r-1)(1+r^1) =$$