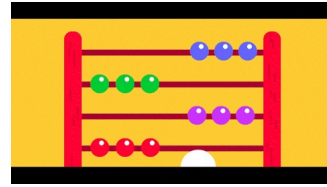


# Introduction to Problem Solving



9:05

- Count of factors
- Prime check
- Sum of first N numbers
- Square root
- Log basics

Tarun Luthra  
Software Dev @  
Scale1

# Count of factors

Given a number N, count its factors.

$N = 24 \rightarrow 1, 2, 3, 4, 6, 8, 12, 24 \rightarrow 8 \text{ factors}$

$N = 10 \rightarrow 1, 2, 5, 10 \rightarrow 4$

Quiz 1

## Brute Force

```
int countFactors(int N) {
```

$\text{count} = 0$

$\text{for } (i \rightarrow 1 \text{ to } N) \{$

$\text{if } (N \% i == 0)$

$\text{count} = \text{count} + 1$

$\}$

$\text{return count}$

```
}
```

$\rightarrow N \text{ iterations}$

## Java

```
static int countFactors(int n) {  
    int ans = 0;  
  
    for (int i = 1; i <= n; i++) {  
        if (n % i == 0) {  
            ans++;  
        }  
    }  
  
    return ans;  
}
```

## Python

```
def countFactors(n):  
    ans = 0  
    for i in range(1, n + 1):  
        if n % i == 0:  
            ans += 1  
    return ans
```

## Is this good?

## Quiz 2

N iterations

Assumption  $\rightarrow 10^8$  iterations  $\rightarrow 1$  second

N	iterations	Execution time
$10^8$	$10^8$ iterations	1 sec
$10^9$	$10^9$ iterations	10 sec
$10^{18}$	$10^{18}$ iterations	

$10^8$  iterations  $\rightarrow 1$  sec

1 iteration  $\rightarrow \frac{1}{10^8}$  sec

$10^9$  iteration  $\rightarrow 10^9 \times \frac{1}{10^8}$  sec = 10 sec

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

=  $10^{10}$  sec

$\approx 317$  years

You  $\rightarrow$  Children  $\rightarrow$  Grandchildren  $\rightarrow 4^{\text{th}} \rightarrow 5^{\text{th}} \rightarrow 6^{\text{th}}$

## Optimising

$i * j = N \rightarrow i \text{ \& } j \text{ are factors of } N$

$\Rightarrow \boxed{j = N/i} \rightarrow i \text{ \& } N/i \text{ are factors of } N$

$N = 24$

$i$	$N/i$
1	24 +2
2	12 +2
3	8 +2
4	6 +2
<hr/>	
6	4
8	3
12	2
24	1

8 factors

$$i \leq \frac{N}{i}$$

$$\Rightarrow i^2 \leq N$$

$$\Rightarrow i \leq \sqrt{N}$$

$N = 100$

$i$	$N/i$
1	100 +2
2	50 +2
4	25 +2
5	20 +2
10	10 +1
<hr/>	
20	5
25	4
50	2
100	1

9

Range - 1 to  $\sqrt{N}$

# Optimised code

```
int countFactors(int N) {
```

*count = 0*

*for (i=1 ; i <=  $\sqrt{N}$  ; i++) {*

*if (N % i == 0) {*

*// i is a factor of N*

*if (i == N/i)*

*count = count + 1*

*else*

*count = count + 2*

*}*

*}*

}

## Java

```
static int countFactorsOptimized(int n) {
    int count = 0;

    for (int i = 1; i * i <= n; i++) {
        if (n % i == 0) {
            if (i * i == n) {
                count++;
            } else {
                count += 2;
            }
        }
    }

    return count;
}
```

## Python

```
def countFactorsOptimized(n):
    count = 0
    i = 1

    while i * i <= n:
        if n % i == 0:
            if i * i == n:
                count += 1
            else:
                count += 2
            i += 1

    return count
```

Iterations =  $\sqrt{N}$

if  $N = 10^{18} \rightarrow 10^9$  iterations  $\rightarrow 10$  sec

Previous  
code



Now

317 yrs

10 sec

# Prime Check

Given N, check if it is a prime number.

That has exactly two factors

Quiz 3

10, 11, 23, 2, 25, 27, 31  $\Rightarrow$  4

1  $\Rightarrow$  Neither prime nor composite

```
bool checkPrime(int N) {
```

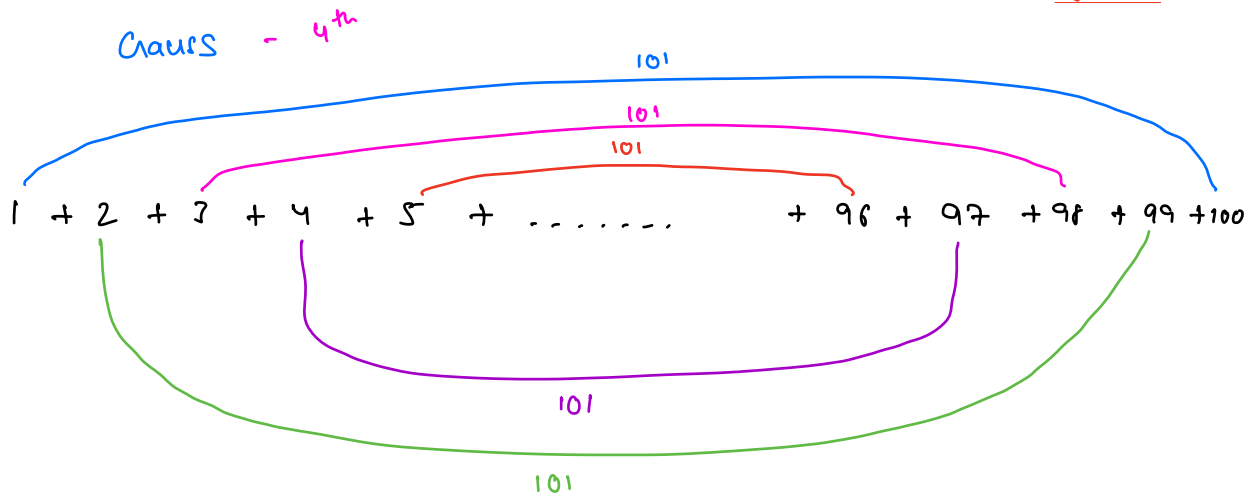
```
    if (countFactors(N) == 2)
        return true
```

```
    else
        return false
```

```
}
```

# Sum 1 - 100

Quiz 4



Sum of every pair = 101  
pairs - 50

$$\Rightarrow \text{Ans} = 101 \times 50 \\ = 5050$$

Sum of first  $N$  natural numbers

Quiz 5

$$\frac{N(N+1)}{2}$$

$$\frac{100(101)}{2} = 5050$$

Break till 10:14 PM

# Square Root

Given a number N, find its square root.

$$N = 25 \rightarrow 5$$

$$N = 100 \rightarrow 10$$

$$N = 36 \rightarrow 6$$

$$N = 30 \rightarrow \text{invalid input}$$

Assume, for this question, we only get perfect squares

## Quiz 6

35, 24, 49, 50

$7^2$

## Quiz 7

a) N

b)  $\sqrt{N}$

c)  $\log N$

d) None

```
int squareRoot(int N) {
```

```
    for (i=1; i<=N; i++) {
```

```
        if (i*i == N)
```

```
            return i
```

```
    }
```

```
}
```

## Java

```
static int sqrt(int n) {
    for (int i = 1; i * i <= n; i++) {
        if (i * i == n) {
            return i;
        }
    }
}
```

## Python

```
def sqrt(n):
    for i in range(n):
        if i * i == n:
            return i
```



Q. Follow up question

what if  $N$  is not a perfect square

$$N=50 \Rightarrow 7$$

$$N=20 \Rightarrow 4$$

$$N=30 \Rightarrow 5$$

$$N=10 \Rightarrow 3$$

Integer part

$\text{floor}(\text{sqrt})$

$$\text{floor}(5.6) \rightarrow 5$$

$$\text{floor}(1.9) \rightarrow 1$$

$$\text{floor}(2.15) \rightarrow 2$$

$$\text{floor}(2.99) \rightarrow 2$$

```
int sqrt(N) {
```

```
    ans = 0
```

```
    for (i=1; i<=N; i++) {
```

```
        if (i*i <= N)
```

```
            ans = i
```

```
    }
```

```
    return ans
```

```
}
```

$N=10$

i	ans = 0
1	1
2	2
3	3
4	3
5	3
6	3
7	3
8	3

Iterations -  $N$  iterations  
ToDo  $\leftarrow$  After optimisations  $\rightarrow \sqrt{N}$

Best algo to find  $\sqrt{N}$

$\rightarrow$  Binary Search -  $\log_2 N$   
 $\rightarrow$  Advanced module

# Logarithm

$$\log_{10} 10,000 = ?$$

Power / Exponent  
↓  
3  
2 = 8  
↑  
Base

Value

$$\log_2 8 = 3$$

To what power should I raise 2 to get the result 8?  $\Rightarrow 3$

Eg  $\log_2 64 = 6$

$$2^n = 64$$

$$2^n = 2^6$$

$$\underline{n = 6}$$

Eg  $\log_3 27 = 3$

$$3^n = 27 = 3^3$$

Eg  $\log_5 25 = 2$

$$5^1 = 25$$

$$5^2 = 25$$

Eg  $\log_2 32 = 5$

$$\log_2 2^{10} = 10$$

$$\log_3 3^6 = 6$$

$$\log_a a^N = N$$

$$2^k = N$$
$$\log_2 N = k$$

Default base

Maths -  $e = 2.73 \dots$

Maths (for calculations) = 10

Computer Science = 2

## Question

Given a positive integer  $N$ , how many times do we need to divide it by 2 until it reduces to 1.

$$N=16$$

$$16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1 \quad \underline{4 \text{ times}}$$

$$N=9$$

Quiz 8

$$9 \rightarrow 4 \rightarrow 2 \rightarrow 1 \quad \underline{3 \text{ times}}$$

$$N=27$$

Quiz 9

$$27 \rightarrow 13 \rightarrow 6 \rightarrow 3 \rightarrow 1 \quad 4 \text{ times}$$

$$\frac{N}{2^0} \xrightarrow{1} \frac{N}{2^1} \xrightarrow{2} \frac{N}{2^2} \xrightarrow{3} \frac{N}{2^3} \xrightarrow{4} \frac{N}{2^4} \xrightarrow{5} \frac{N}{2^5} \dots \xrightarrow{k} \frac{N}{2^k}$$

$$\frac{N}{2^k} = 1$$

$$\Rightarrow 2^k = N$$

$$\Rightarrow \log_2 N = k$$

$\log_2 N$  steps

# Intermediate Content

## ✓ • Introduction to Problem Solving

- Time Complexity 1, 2
- Arrays - 6
  - Array Problems
  - Prefix Sum
  - Carry Forward
  - Subarrays
  - Sliding Window
  - 2D Arrays

2 Months

## ✓ • Interview Problems 2

- Modular Arithmetic ✂
- Bit Manipulations - 2
- Sorting
- Strings
- Hashing - 2
- Recursion - 2
- Linked Lists
- Trees Basics

# Doubts

Thank  
you

MWF - 9 - 11:30 PM

math module → log functions

Doubts → TAs  
→ Ask for peers  
→ Unlock hints  
→ Doubt Session

Chess, Sudoku, Puzzles → youtube - TedEd Riddles

First priority - Assn / HW

→ [hackerrank.com](https://www.hackerrank.com)

→ [hackerearth.com](https://www.hackerearth.com)

Interview

→ [leetcode](https://leetcode.com)

→ [interviewbit](https://www.interviewbit.com)

Read first Java / Python

Good  
Night

Thank  
You

Monday

