



Today's agenda

- ↳ No. of factors
- ↳ Prime numbers
- ↳ Sum of N natural nos
- ↳ floor & ceil
- ↳ $\text{Sqrt}()$



AlgoPrep



Q) Count no. of factors:

↳ Given a number N , find the Count of factors.

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

$N = 36 \rightarrow \{1, 2, 3, 4, 6, 9, 12, 18, 36\} \rightarrow 9$

```
public static void main ( ) {  
    Scanner scn = new Scanner (System.in);  
    int n = scn.nextInt();  
    int Count = 0;  
    for (int i = 1; i <= n; i++) {  
        if (n % i == 0) {  
            Count++;  
        }  
    }  
    return Count;  
}
```

1 sec = 10^8 iterations.

No. of iterations = N times

$N = 10^9$

10^9 iterations

→ How many secs will it take?

1 sec = 10^8 iteration

$\frac{1}{10^8}$ secs = 1 iteration

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

10^1 iterations = 10 secs

$N = 10^{18} \rightarrow$ Seconds??

10^8 iteration = 1 sec

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

10^{18} iterations = $\frac{1}{10^8} * 10^{18} = 10^{10}$ sec

10^{10} secs = 317 years

you → child → grandchild → 3rd → 4th → 7th gen.



//optimize

$$\hookrightarrow i * j = N \Rightarrow j = N/i$$

$\rightarrow i$ is one factor \rightarrow other factor is $N/i \text{ or } j$

$N = 24$			$N = 36$		
i		$j = N/i$	i		N/i
1	<	24	1	<	36
2	<	12	2	<	18
3	<	8	3	<	12
4	<	6	4	<	9
6	>	4	6	=	6
8	>	3	9		4
12	>	2	12		3
24	>	1	18		2
			36		1



// Pseudo code

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

→ No. of iterations → \sqrt{n}

$n = 10^{18}$ → Seconds??

↳ 10^8 iterations → 1 sec ⇒ 1 iteration = $\frac{1}{10^8}$ sec

↳ $\sqrt{10^{18}} = 10^9$ iterations

10^9 iterations = $\frac{1}{10^8}$ sec * $10^9 = 10$ sec.

→ 317 yrs → 10 sec



$N=24$

$\rightarrow \sqrt{24} \rightarrow 4.9$

$\rightarrow \text{count} = 0 + 2 + 2 + 2 + 2$

i	$i \leq \sqrt{n}$	$N \% i == 0$	
1	+	+	24
2	+	+	12
3	+	+	8
4	+	+	6
5			
	6		

$\hookrightarrow \text{exit}$

$N=36$

$\text{Count} = 0 + 2 + 2 + 2 + 2$

```
int Count = 0;
for (int i = 1; i <= sqrt(n); i++) {
    if (N % i == 0) {
        Count = Count + 2;
    }
}
```

i	$i \leq \sqrt{n}$	$N \% i == 0$	N/i
1	+	+	36 +2
2	+	+	18 +2
3	+	+	12 +2
4	+	+	9 +2
5	+		
6	+	+	6 +1

$\text{return Count};$



$N = 36 \rightarrow \sqrt{N} = 6$
 $count = 0 + 2 + 2 + 2 + 2 + 1$

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

i	$i \leq \sqrt{n}$	$n \% i == 0$	n/i
1	t	t	36
2	t	t	18
3	t	t	12
4	t	t	9
5	t	f	
6	t	t	6
7			

6 exit



AlgoPrep



Q) Prime numbers

↳ Given a number n , check if the number is a prime no.

```
void PrimeNo (int n) {  
    int count = 0;  
    for (int i = 1; i <= n; i++) {  
        if (n % i == 0) {  
            if (i != n/i) { count = count + 2; }  
            else { count = count + 1; }  
        }  
    }  
    if (count == 2) { s.o.p("Prime"); }  
    else { s.o.p("Not Prime"); }  
}
```



Quiz 1: Sum of all the numbers from 1 to 10.

$$\hookrightarrow \frac{10 * 11}{2} = 55$$

Quiz 2: Sum of all the numbers from 1 to 1000.

$$\hookrightarrow \frac{1000 * 1001}{2} = 500500$$

Q) Sum of first N natural numbers.

whole no: 0, 1, 2, ...

→ Gauss (4th class)

$$S = 1 + 2 + 3 + \dots + 998 + 999 + 1000$$

$$S = 1000 + 999 + 998 + \dots + 3 + 2 + 1$$

$$2 * S = 1001 + 1001 + 1001 + \dots + 1001 + 1001 + 1001$$

$$2 * S = 1001 * 1000$$

$$S = \frac{1001 * 1000}{2} = 500500$$

// Sum of first N natural numbers.

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

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

$$2 * S = (N+1) + (N+1) + (N+1) + \dots + (N+1) + (N+1) + (N+1)$$

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



Quiz 3: Sum of 1st N whole numbers.

$$0 + 1 + 2 + \dots + N - 1$$

\Downarrow

$$1 + 2 + \dots + N - 1 \Rightarrow \frac{N(N-1)}{2}$$

Sum of 1st N whole numbers = Sum of 1st $N-1$ natural numbers

Sum of first 5 whole numbers?

$$0 + 1 + 2 + 3 + 4$$

Break till 9:30 pm



// floor(num) → just smaller or equal integer

Ex: 7.4 → 7

8.9 → 8

100.01 → 100

90 → 90

20.99 → 20

3 → 3

→ math.floor(num) → ⌊



AlgoPrep



$\text{Ceil}(\text{num}) \rightarrow$ just greater or equal integer

Ex: $7.4 \rightarrow 8$
 $8.9 \rightarrow 9$
 $100.01 \rightarrow 101$
 $90 \rightarrow 90$
 $20.99 \rightarrow 21$
 $3 \rightarrow 3$



AlgoPrep



Q) Given N , return $\text{floor}(\text{sqrt}(N))$.

ex: $N=60 \rightarrow 7.7 \rightarrow 7$

$N=31 \rightarrow 5.5 \rightarrow 5$

$N=29 \rightarrow 5.2 \rightarrow 5$

$N=16 \rightarrow 4.0 \rightarrow 4$

```
int sqrt(int n) {
```

```
    int ans = 1;
```

```
    int i = 1;
```

```
    while (i*i <= n) {
```

```
        ans = i;
```

```
        i++;
```

```
    }
```

```
    return ans;
```

```
}
```

$N=60$

i	$i*i \leq N$	ans
1	+	1
2	+	2
3	+	3
4	+	4
5	+	5
6	+	6
7	+	7
8	x	x
...		



Tracing

$N = 24$

```
int sqrt(int n) {
```

```
    int ans = 1;
```

```
    int i = 1;
```

```
    while (i*i <= n) {
```

```
        ans = i;
```

```
        i++;
```

```
    }
```

```
    return ans;
```

→ i $i*i \leq N$ ans

1 t 1

2 t 2

3 t 3

4 t 4

5 f

break

2

AlgoPrep



Q) Given an integer N , return the difference between the product of digits and Sum of its digits.

$N: 234 \rightarrow$ Product of digits: $2 \times 3 \times 4 = 24$

Sum of digits: $2 + 3 + 4 = 9$

\Downarrow

$$24 - 9 = 15$$

int Sum = 0;

int multiply = 1;

while (n > 0) {

int lastdigit = n % 10;

Sum = Sum + lastdigit;

multiply = multiply * lastdigit;

n = n / 10;

return multiply - Sum;

$N: 234 \rightarrow 23 \rightarrow 20$

Sum = 0

multiply = 1

n > 0

lastdigit

Sum

multiply

+

4

4

4

+

3

7

12

+

2

9

24

\downarrow

break