4/1/2023

Introduction to problem solving

Start at 9:05 pm

2014 → IIIT Allahabad

→ 2 months → intermediate
→ 4 months → advanced dsa

→ 3 classes / week (M, W, F) 9-11 pm

→ Pseudo code → language agnostic.

→ English

→ Revise lecture [1]
→ Assignments [2] ——→ 6 months
→ Homework [3]

LLD, HLD, springboot, react

Q1 Count of factors.

↓

any no. that divides N completely.

`N%i==0` , then i is a factor of N.

Eg1 N = 24 [1, 2, 3, 4, 6, 8, 12, 24] ⇒ ans = 8

Eg2 N = 10 [1, 2, 5, 10] , ans = 4

Brute force → correct solution but with no optimization.

Pseudo code

```
int countfactors (int N) {                    → 10^10

    int ans = 0;
    for (int i = 1; i <= N; i++) {            → N

        if (N % i == 0) {
            ans++;
        }
    }

    return ans;
}
```

Assumptions

10^8 iterations $\Rightarrow$ 1 sec.

N	Iterations	Time taken
10^8	10^8	1 sec.
10^{10}	10^{10}	100 sec.
10^{18}	10^{18}	10^{10} sec.

317 years =

$10^8 \rightarrow 1$ sec

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

$10^{10} \rightarrow \frac{1}{10^8} \times 10^{10} \Rightarrow 100$ sec $\Rightarrow$

OBSERVATIONS → (Key to problem solving)

N : 'i' is a factor
=
$i \times j = N$

$\boxed{j = N/i}$

Observation 1 - factors come in pairs.

N = 24

	i	N/i	Count
→	1	24	2
→	2	12	4
→	3	8	6
→	4	6	8

$i < N/i$

$\boxed{i \leq N/i}$

1 → 24

$\boxed{24/1} \Rightarrow 24$

$24/2 \Rightarrow 12$

→	6	4
	8	3
	12	2
	24	1

$i, N/i$

$i \times N/i \Rightarrow N$

$i \leq N/i$

$i^2 \leq N$

$i \leq \sqrt{N}$

$i \rightarrow N$

$N \% i == 0$

$i \times X = N$

$2 \times 4 \Rightarrow 8$

$i \quad j \quad N$

$j = N/i$

$\rightarrow j$

8

1 2 4 8

$i \times N/i = N$

$N = 100$

i	N/i	count
1	100 ✓	2
2	50 ✓	4
4	25 ✓	6
5	20 ✓	8
10 ✓	10 ✓	10 → 9
20	5	
25	4	
50	2	
100	1	

$i \leq N/i$

$i < N/i$

$i \leq \sqrt{N}$

$\sqrt{100} \Rightarrow 10$

10×10

$i \leq N/i$ $\quad$ $i \leq \sqrt{N}$

Pseudo code →

```
int countFactors (int N) {
    int ans = 0;

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

        if (N % i == 0) {
            if (N/i == i)
                ans++;
            else
                ans += 2;
        }
    }
    return ans;
}
```

Total iterations → $\sqrt{N}$

$i \mid N / i$

$i \mid N / i$

$i \mid N / i$

$i^2 \le N$

$i \le \sqrt{N}$

$\sqrt{24}$

4

1	2	3	4
↓	↓	✗	✗
24	12	8	6

10^8 it → 1sec

N	iteration	Time taken
10^{18}	10^9	10 seconds.

$\sqrt{10^{18}} = 10^9$, $\sqrt{10^9 * 10^9} \Rightarrow \boxed{10^9}$

Q2 Given a no. N, return true if it is a prime no

2 [1,2]
3 [1,3]
4 x [1,2,4]
5 [1,5]
?

⇓

count of factors = 2

Pseudo code

```
bool checkPrime ( int n) {

    int count = countFactors (n);
    if (count == 2)
        return true;

    return false
}
```

$S = 1 + 2 + 3 + 4 + \ldots + 100$

$S = 100 + 99 + 98 + 97 + \ldots + 1$

$2S = 101 + 101 + 101 + 101 + \ldots + 101$

$2S = 101 \times 100$

$S = \underline{101 \times 100} = 5050$

2

* Sum of 1st n natural nos.

$S = 1+2+3 \cdots \cdots + n$

$S = n+(n-1)+(n-2) \cdots + 1$

$2S = (n+1)+(n+1) \cdots \cdots + (n+1)$

$2S = (n+1)\,n$

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

Q-3 Given a perfect square, N. Find the square root of N.

Pseudo code

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

① $\sqrt{N}$ ② N
③ log N ④ NOT

N → [1, N]

N → $\sqrt{N}$ → i

$\sqrt{N} \times \sqrt{N} = N$

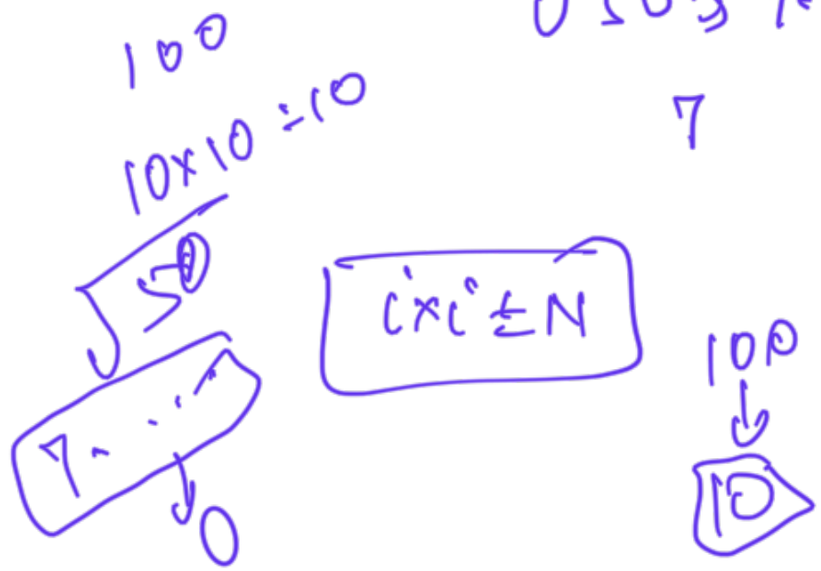
i * i = N

Q-4 Find floor (sqrt(N)), N is not always a perfect square.

floor (x) ⇒ greatest integer ≤ x

Eg 1 floor(3) → 3
floor (2.5) → 2

N = 50 , ans = 7

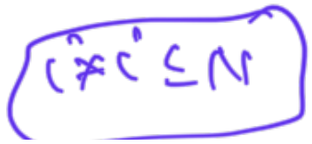


√50 ⇒ 7.....

7

Pseudo code

```
int findRoot(int n) {
    int ans = 1;
    for (int i = 1; i ≤ N; i++) {
        if (i*i ≤ N) {
            ans = i;
        }
        else {
            break;
        }
    }
    return ans;
}
```



```
        ans = i;
    }
    else
        break;
}
return ans;
}
```

$i * i > N$

i	i * i	ans
1	1	1
2	4	2
3	9	3
4	16	4
5	25	5
6	36	6
7	49	7
8	64	

N = 50

$i * i > N$

$i - 1$

$\frac{10^a}{10^b} \Rightarrow 10^{a-b}$, $10^a \times 10^b = 10^{a+b}$

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

$10^{18} \Rightarrow \frac{1 \times 10^{18}}{10^8}$

$\frac{10^{18}}{10^8} \Rightarrow 10^{18-8} \Rightarrow 10^{10}$