N Satya Sai Siva Rama Krishna

2019 CSE graduate

2 years Teaching Experience

Given $N >= 1$, when is it Prime?

$N == 1$ { Neither prime nor Composite }

- If N is prime: If 1 & itself are factors : x
- If N has factor == 2, N is prime

$N >= 1$

```
bool IsPrime(N) { // N iterations
    int c = 0
    for(i = 1; i <= N; i++){
        if(N % i == 0){
            c++
        }
    }
    if(c == 2){
        return True
    }
    else return False
}
```

Assumption: 10^8 iterations per sec

If $N = 10^7$, 10^7 iterations

10^8 iterations = 1 sec

10^9 iterations = 10 sec

If $N = 10^{18}$, 10^{18} iterations

10^8 iterations = 1 sec

10^{18} iterations = 10^{10} sec

≈ 317 years

You → Your kids → Grand kids → 4th Gen → 5th generat → 6th gener

✗ ✓ ✓ ✗ ✓ ✓

// Optimization

// a, b, N all are Integers

$a * b = N$, $b = N/a$

{a, b} are factors of N

{a, N/a} are factors of N

If a is factor

: N/a is also factor

Relational Operator

<, >, <=, >=, !=, ==

If $a <= 57$

Max a = 57

N = 24 Cut

i	N/i	
1	24	+2
2	12	+2
3	8	+2
4	6	+2
6	4	8
8	3	
12	2	
24	1	

Part I

$i <= N/i$

$i^2 <= N$

$i <= \sqrt{N}$

[1, $\sqrt{N}$] & get all factors

N = 100 Cut

i	N/i	
1	100	+2
2	50	+2
4	25	+2
5	20	+2
10	10	+1
20	5	9
25	4	
50	2	
100	1	

```
bool isPrime(N) {
    int c = 0 -> [1 √N]
    for(i = 1; i*i <= N; i++){
        if(N % i == 0){
            // i is factor N/i is also factor
            if(i == N/i){
                c = c + 1
            }
            else{
                c = c + 2
            }
        }
    }
    if(c == 2)
        return true
    else return false
}
```

Iterations $\sqrt{N}$

$N = 10^{18}$

$\sqrt{10^{18}}$

$\rightarrow 10^9$ iterations = 1sec

//

Sum of N Natural Numbers = $\frac{(N)(N+1)}{2}$

Murugan

$S = 1 + 2 + 3 + 4 + 5 + 6 + \cdots\cdots + 97 + 98 + 99 + 100$

$S = 100 + 99 + 98 + 97 + 96 + 95 + \cdots\cdots + 4 + 3 + 2 + 1$

$2S = 101 + 101 + 101 + 101 + 101 + 101 + \cdots\cdots + 101 + 101 + 101 + 101$

$2S = (100)(101)$ $S = (50)(101)$

$S = 1 + 2 + 3 + 4 + \cdots\cdots + N-2 + N-1 + N$

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

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

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

→ Sum of N Whole Numbers

= $0 + 1 + 2 + 3 + \cdots\cdots N-1$

= Sum (N-1) Natural Numbers

= $\frac{(N-1)(N)}{2}$

→ 2 Whole : {0, 1}

→ 3 Whole : {0, 1, 2}

C Iteration

389) Given N, Number of times we need to divide by 2 till it reaches (1) int $(\log_2 N)$

$\log_2 2^{10} = 10$, $\log_a a^n = n$

Note: / → Integer Quotient

$7/2 \to 3$ $4/2 \to 2$ $9/2 \to 4$

$\log_2 2^5 = 5$,

N = 1 : 0

N = 2 : $2/2 \to 1$: 1

2^1 ⟶ $\log_2 2 = 1$

N = 4 : $4/2 \to 2/2 \to 1$: 2

2^2 ⟶ $\log_2 4 = 2$

N = 8 : $8/2 \to 4/2 \to 2/2 \to 1$: 3

2^3 ⟶ $\log_2 8 = 3$

N = 15 : $15/2 \to 7/2 \to 3/2 \to 1$: 3

⟶ $\log_2 15 = 3.90$

N = 16 : $16/2 \to 8/2 \to 4/2 \to 2/2 \to 1$: 4

2^4 ⟶ $\log_2 16 = 4$

N = 27 : $27/2 \to 13/2 \to 6/2 \to 3/2 \to 1$: 4

⟶ $\log_2 27 = 4.75$

N = 50 → $\log_2 50 = 5.64$

$50/2 \to 25/2 \to 12/2 \to 6/2$

$\downarrow$

$3/2$

$\downarrow$

1

Note

5 times

10:33 break

Note

1) Sum of A.P / G.P
2) $\log_2^N$ value

} Reference material

→ Amazon PICO's

Given a perfect Square $N >= 1$, find Sqrt(N)

N = 25 → 5

N = 100 → 10

N = 40 } x Invalid Input

// N is perfect Square $N >= 1$

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

→ i = $\sqrt{N}$ it will stop:

i → [1, $\sqrt{N}$]: {$\sqrt{N}$} → $\sqrt{N}$ Iterations

Iterations

N = 36

[1 6]

i =	1	2	3	4	5	6
	x	x	x	x	x	return 6

N = [1 N] from 1 to N sqrt of N is always present

100 → [1 100] from 1 to 100 sqrt(100) is present

centre

[1 100] → 50 * 50 > 100 → 51, 52, 53, -- 100

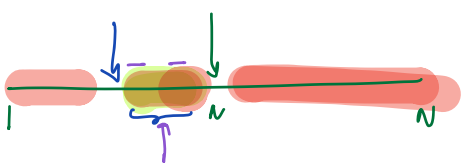
[1 49] → 25 * 25 > 100 x { 26, 27, -- 49 }

[1 24] → 12 * 12 > 100 x { 13, 14, 15, .. 24 }

[1 11] → 6 * 6 < 100 x { 1, 2, 3, 4, 5 }

[7 11] → 9 * 9 < 100 x { 7, 8 }

[10 11] → 10 * 10 = 100



For Example

- [1, 2, 3 . . . N]
- At every iteration we neglect N/2

N → N/2 → N/4 → N/8 → . . . 1 : ≈ {log2 N iterations}

At max

You will get sqrt(N)

$\sqrt{N}$ vs $\log_2 N$

$N = 2^{30}$ — 2^{15} — 30

$N = 2^{60}$ — 2^{30} >> 60

Content Intermediate : 2 Months

1) Time Complexity & Space Complexity
2) Arrays : 6
   - → Introduction Arrays
   - → Prefix Sum
   - → Carry forward
   - → Subarrays / Sliding Window / Contribution Technique
   - → 2D Matrix
   - → Interview Problems
3) Bits Manipulations : 2 →
4) Maths & Arrays : 2 →
5) Sorting / Strings / Hashmap : 4
6) Recursion : 3 →
7) Subset / Subsequ
8) Linked List Basics

Session Expectations :

1) 9 PM - 7:05 PM
2) 11:30 PM :
3) Doubts : 11:30 PM -
4) Conceptual Doubts
   → During session :
   → In Doubt Session
5) Give your best to attend live Session.
6) Revision & Notes
7) Language Agnostic

Assignments

: Per week : 20 Problems
Assignment + HW

: 8 weeks : 160 Problems

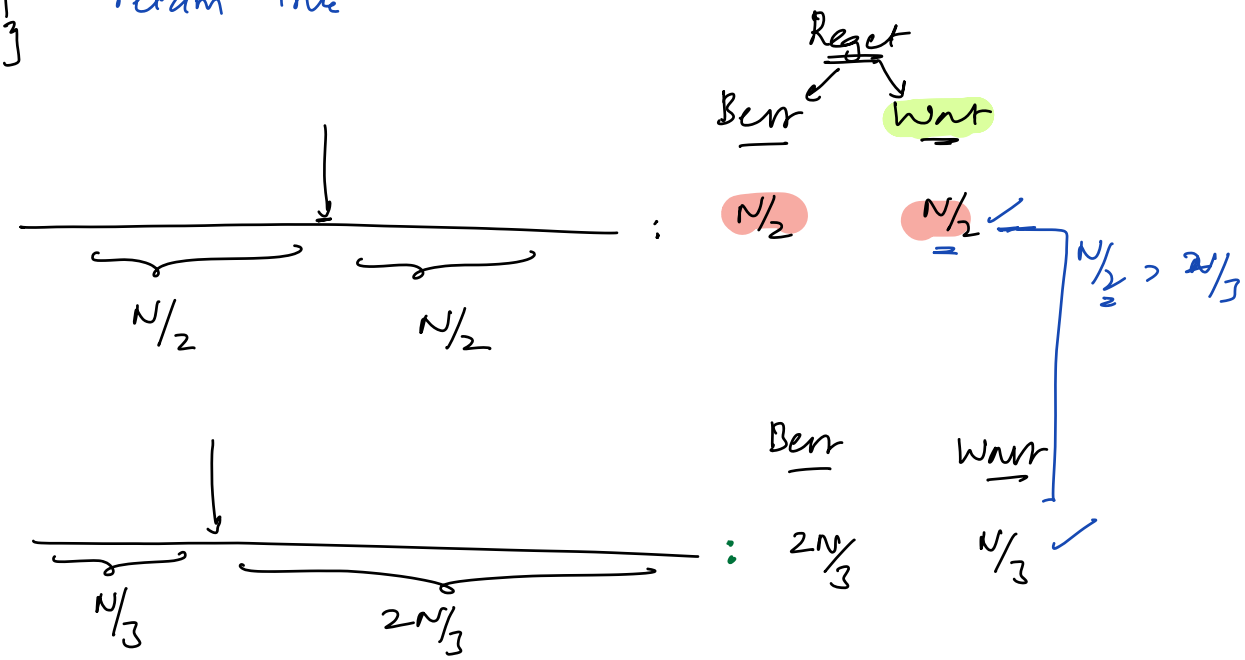
Doubts: Assignm

1) Debug on your own
2) Teaching Assistant (TA)
3) Ask me :

Problem Solving Session
→ will discuss HW/ Assignment Problem

Doubt:

```
isPrime(N) {
    for(i=2; i*i<=N; i++) {
        if(N%i==0) return false
    }
    return True
}
```



⇒ log basics } =

⇒ GP basic : Sum of GP

⇒