

Let's start at 8:02 PM

L49

Introduction to Sieve of Eratosthenes

Join Discord - <https://bit.ly/ly-discord>

RECAP

Number Theory Starts

Some Basic Terminologies

(I) Integers $\Rightarrow \dots -10, -9, -8, \dots -1, 0, 1, 2, \dots 5 \dots$

(W) Whole numbers $\Rightarrow 0, 1, 2, 3, \dots$

(N) Natural numbers $\Rightarrow 1, 2, 3, \dots$

Composite

Prime.

given a ^{fixed} number N , $\sum_{d|N} d^2 \Rightarrow 1, 2, 3, 4, 6, 8, 12, 24$

24
 \swarrow

$| \Rightarrow$ divides

d divides $N \Rightarrow d$ is a factor of N ($n \times d = 0$)
 N is a multiple of d .

Fundamental Theorem of Arithmetic

$$N = 24$$

$$\hookrightarrow 2^3 * 3^1$$

$$N = 47$$

$$\hookrightarrow 47$$

Every +ve integer can be uniquely represented
as a product of primes.

Let's do a puzzle:

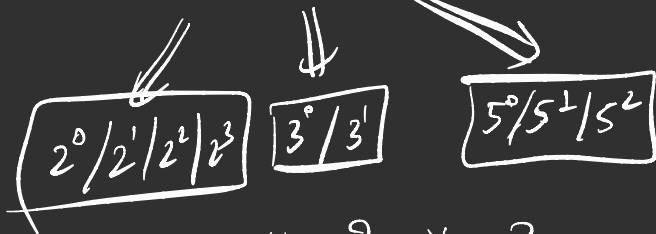
Given a number, can we find a formula to find its number of divisors?

$$N = p_1^{\alpha_1} \times p_2^{\alpha_2} \times p_3^{\alpha_3} \dots p_k^{\alpha_k}$$

$$N = p_1^{\alpha_1} * p_2^{\alpha_2} * p_3^{\alpha_3} \dots p_k^{\alpha_k}$$

$$d(N) = (\alpha_1 + 1) * (\alpha_2 + 1) * (\alpha_3 + 1) \dots * (\alpha_k + 1)$$

$$600 = 2^3 * 3^1 * 5^2$$



$2^0/2^1/2^2/2^3$

$3^0/3^1$

$5^0/5^1/5^2$

$$4 * 2 * 3 = 24$$

Finding if a number is prime

```
if (n == 1) return false;  
for (i = 2; i < n; ++i) {  
    if (n % i == 0)  
        return false;  
}  
return true;
```

Time $\Rightarrow O(N)$

```
if (N == 1) return false;  
for (i = 2; i * i <= N; ++i)  
    if (N % i == 0)  
        return false;  
  
return true;
```

Time $\Rightarrow O(\sqrt{N})$

$$\begin{array}{c}
 (a < N) \quad (b < N) \\
 N = \underbrace{(a) \times (b)}_{\substack{\downarrow \quad \downarrow \\ > \sqrt{N} \quad > \sqrt{N}}}
 \end{array}$$

How can their product be equal to N ?

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

Doing the prime factorization of a number

$$600 \Rightarrow \{2:3, 3:1, 5:2\}$$

$$i=2, 3, 4, 5, 6, \dots$$

$$i=5, C=2$$

$$N=600$$

$$300$$

$$150$$

$$75$$

$$25$$

$$5$$

$$1$$

$$N \approx 11$$

2, 3, ~~4~~

map<int, int> pFact (int n) {

map ans;

for (i=2; i*i <= n; ++i) {

if (n%i)

continue;

int c=0;

while (n%i==0)

n/=i, c++;

ans.put(i, c);

if (n>1) ans.put(n, 1);

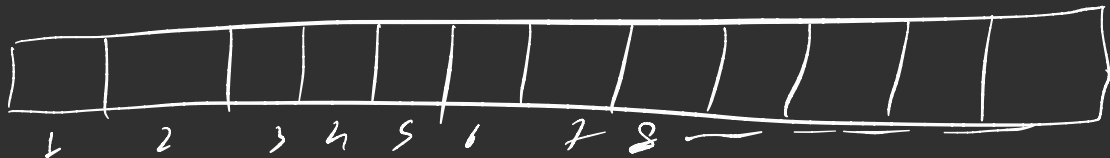
return ans;

Time = $O(\sqrt{N})$

Sieve of Eratosthenes : Introduction

$N \approx 100$

isPrime.



Let's visualise

Intuition

Pseudo-Code

What's the time taken?

$$N + \frac{N}{2} + \frac{N}{3} + \frac{N}{4} \dots \frac{N}{N}$$

$$\Rightarrow N * \underbrace{\left(1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} \dots \frac{1}{N} \right)}_{\log_2 N}$$

$$\begin{array}{ccc}
 \frac{1}{1} + \underbrace{\frac{1}{2} + \frac{1}{3}}_{2 \text{ terms}} + \underbrace{\frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \frac{1}{7}}_{4 \text{ terms}} + \underbrace{\frac{1}{8} + \frac{1}{9} + \frac{1}{10} + \frac{1}{15}}_{8 \text{ terms}} \\
 \Downarrow \qquad \qquad \qquad \Downarrow \qquad \qquad \qquad \Downarrow \\
 \leq 1 \qquad \qquad \qquad \leq 1 \qquad \qquad \qquad \leq 1
 \end{array}$$

$$1 + 1 + 1 + \dots + 1 \Rightarrow (\leq \log N)$$

Some optimisations?

Let's implement

Improved Time Complexity

$$O(N \log \log N)$$

Let's do a problem! (Hai himmat?)

Counting Primes

Thank You!

Reminder: Going to the gym & observing the trainer work out can help you know the right technique, but you'll muscle up only if you lift some weights yourself.

So, PRACTICE, PRACTICE, PRACTICE!