SCALER

Prime Numbers Advanced

TABLE OF CONTENTS

Prime Numbers

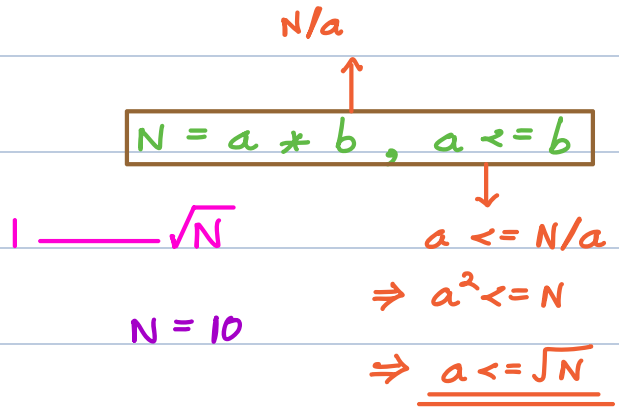
A natural number having 2 factors 1 & itself.

2 3 5 7 11 13 17 . . .

< Question > : Given a number N. Check if it is prime or not.

N = 23 → Ans = true

N = 46 → Ans = false




```
for i → 2 to √N {
    if (N % i == 0)
        return false
}
return true
```

TC = O(√N) SC = O(1)

< Question > : Given an integer N. Check every number from 1 to N if it is a prime number or not.

$1 \le N \le 10^6$

N = 10 [1 2 3 4 5 6 7 8 9 10]

F T T F T F T F F F

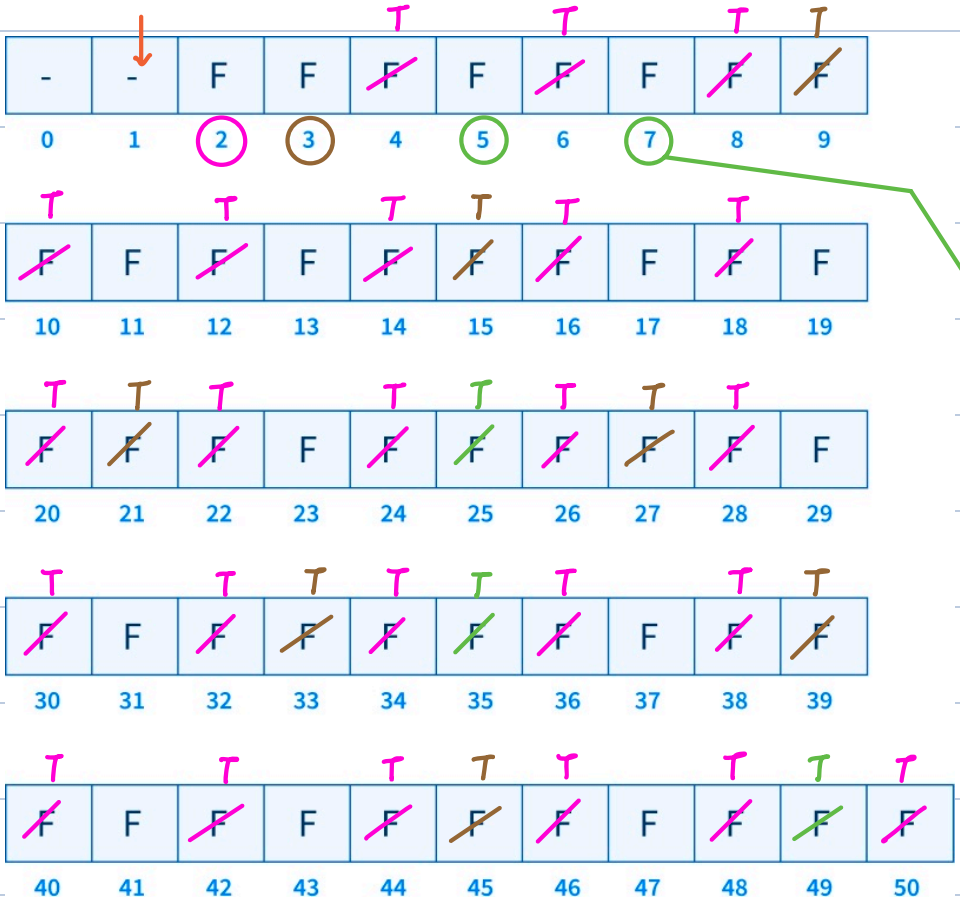
Bruteforce →

```
for i → 1 to N {
    isP[i] = checkPrime(i)
}
```

TC = O(N * √N) SC = O(1)

Sieve of Eratosthenes → chocolates for all except prime no.

class monitor



5 * 2 → 10
5 * 3 → 15
5 * 4 → 20
5 * 5 → 25

7 * 7 = 49

∀ i, isP[i] = true

isP[0] = isP[1] = false

for i → 2 to √N { // for (i = 2; i * i <= N; i++)

    if (isP[i]) {

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

            isP[j] = false

        }

    }

} return isP

SC = O(1) / O(N)

TC

i

2 → 4, 6, . . . N → ~N/2

3 → 9, 12, 15 . . . → ~N/3

4 → ———

5 → 25, 30, 35 . . . → ~N/5

$TC \rightarrow \frac{N}{2} + \frac{N}{3} + \frac{N}{5} + \frac{N}{7} + \frac{N}{11} \cdots$

$= N\left(\frac{1}{2} + \frac{1}{3} + \frac{1}{5} + \frac{1}{7} \cdots\right) <= N\left(\frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{6} \cdots\right)$

reciprocal of prime numbers

$\log(\log(N))$

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

$\sum_{\forall i} \frac{1}{i} \rightarrow \int \frac{1}{x} dx = \log(x)$

$\log(N)$

$O(N \log(N))$

$\log_2(\log_2(2^{\overbrace{2^{32}}^{\sim 10^9}})) = \log_2(32) = 5$

< Question > : Find the count of divisors for every integer from 1 to N.

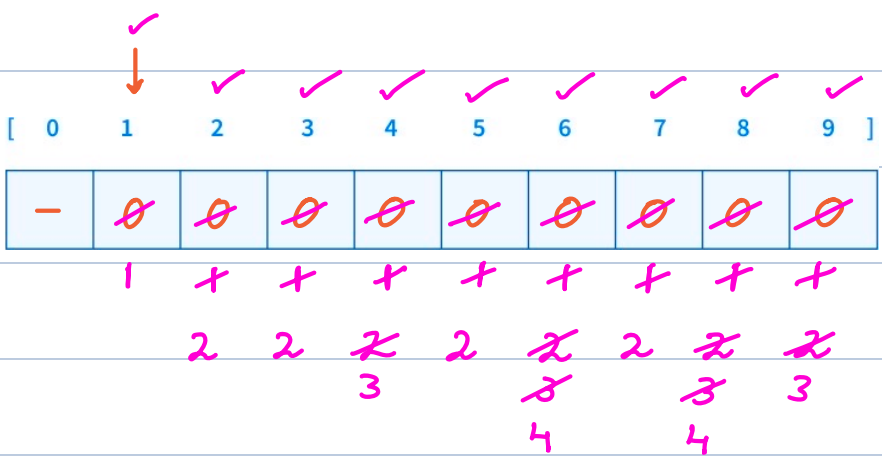
N = ~~10~~ 9

	0	1	2	3	4	5	6	7	8	9
ans[ ] →	–	1	2	2	3	2	4	2	4	3

Bruteforce →

```
for i → 1 to N {
    ans[i] = countDivisors(i)
}
```

TC = O(N √N) SC = O(1)



∀i, ans[i] = 0

```
for i → 1 to N {
    for (j = i; j <= N; j += i) {
        ans[j]++
    }
}
```

TC = O(N log(N)) SC = O(1)

Sorted Permutation Rank

< *Question* > : You are given a string A containing distinct characters (no characters are repeated). The task is to find the rank of this string among all its permutations when sorted in lexicographical (dictionary) order.

s = "cab"

a b c 1
a c b 2
b a c 3
b c a 4
c a b 5 ← Ans
c b a 6

s = "play"

a < l < p < y

a _ _ _ → 3! = 6

l _ _ _ → 3! = 6

p a _ _ → 2! = 2

p l a y → 1

15 (Ans)

s = "date"

a < d < e < t

a _ _ _ → 3! = 6

d a e _ → 1! = 1

d a t e → 1

8 (Ans)

```
fact[0] = fact[1] = 1
for i → 2 to N {
    fact[i] = (fact[i-1] * i) % M
}

rank = 1
for i → 0 to (N-1) {
    ch = s[i]
    crt = 0
    for j → (i+1) to (N-1) {
        if (s[j] < ch) crt++
    }
    rank += crt * fact[N-i-1]
    rank %= M
}
return rank
```

TC = $O(N^2)$ SC = $O(N)$ → fact[]

"utkarsh" (5)

	crt	rank = 1
< 'u' ⇒ 6	6	+ 6*6!
u t k	5	+ 5*5!
	2	+ 2*4!
		⋮