Lec-21

25/02/25 4:47 PM

$2^n + n^3$

10^8

```
for(i=2 i<n i*i>2) → n
if(isprime(i)) → n
count++
```

n^2

Constraints:

- `0 <= n <= 5 * 10^8`

$(5 \times 10)^2 = 25 \times 10^2$

n

```
for(i=2 i<n i*i>2)
{ if c m%i==0>2
  ~ for
}
```

39

2 3 5 6 4

$a^2 = N$

$a = \sqrt{N}$

$n = a \times b$

$a > b$

$a == b$ | $a > b$ | $a < b$

$a == b == \sqrt{n}$

$O(n)$

$n = 10^1$

Sieve of Eratosthenes

0	1	2	3	4	5	6	7	8	9
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

prime numbers:

2×2
2×3
2×4
2×5

$(i \times i)$

$\left[\frac{N}{2} + \frac{N}{3} + \frac{N}{5} + \frac{N}{7} + \frac{N}{11} + \frac{N}{13}\right]$

$N\left[\frac{1}{2} + \frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \frac{1}{11} - - -\right]$

$N \times [\log \log N]$

$N = 1024$

$\log_2^{2^{10}}$

$\log_2 10 \log_2 2$

$\log_2 10 = 3 \cdots$

```
public static int PrimeSieve(int n) {
    int[] prime = new int[n];
    prime[0] = 1;
    prime[1] = 1;
    for (int i = 2; i * i < prime.length; i++) {
        if (prime[i] == 0) {
            for (int j = 2*i * j < prime.length; j++) {
                prime[i * j] = 1;
            }
        }
    }
}
```

$(i \times i > 2)$

2×2
2×3
2×4
2×5
2×6
$2 \times$

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

(i) count all palindromic substr.
(ii) max lam pam
(iii) count odd sum
(iv) cino ema

axis-orbit

orbit

axis

String

odd
even

naan

4
2

0-0=0
0+0=0
0-1=-1
0+1=1
1-0=1
1+0=1
1-1=0
1+1=2

$2-0=2$
$2+0=2$
$2-1=1$
$2+1=3$
$2-2=0$
$2+2=4$
$2-3=-1$
$2+3=5$

4-0=4
4+0=4

N i t i N
0 1 2 3 4

t
i
t
i
n

axis-orbit
axis+orbit

naan

n
a
a
n

4

aa
naan

0.5

0.5-0.5=0
0.5+0.5=1
1.5-0.5=1
1.5+0.5=2
1.5-1.5=0
1.5+1.5=3
1.5-2.5=-1
1.5+2.5=4

N a a N
0 1 2 3
1.5 of

disk-move

src=A

hlp=B

des=C

src=A

hlp=B

des=C

src->hlp

des

```
public static void TOH(int n, String src, String hlp, String des) {
    if (n == 0) {
        return;
    }
    TOH(n - 1, src, des, hlp);
    System.out.println("Move " + n + "th disk from " + src + " to " + des);
    TOH(n - 1, hlp, src, des);
}
```

Move 1th disk from A to C
Move 2th disk from A to B
Move 1th disk from C to B
Move 3th disk from A to C
Move 1th disk from B to A
Move 2th disk from B to C
Move 1th disk from A to C

A B C

TOH(n, src, hlp, des)

TOH(n-1, src, des, hlp)

TOH(n, src, des, hlp)

TOH(n-1, hlp, src, des)

n=1 hlp=B
src=A des=C

n=2
src=B
hlp=A
des=C

n=3
src=A
hlp=C
des=B

main