Lec-20

28/06/25 12:48 PM

N- queen

N=1

4x4

```
public static void print(boolean[][] board, int tq, int row) {
    if (tq == 0) {
        Display(board);
        return ;
    }
    for (int col = 0; col < board[0].length; col++) {
        if (isitsafe(board, row, col)) {
            board[row][col] = true;
            print(board, tq - 1, row + 1);
            board[row][col] = false;
        }
    }
}
```

Already done

sieve of Eratosthenes

										Prime numbers
	2	3	4	5	6	7	8	9	10	
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	
101	102	103	104	105	106	107	108	109	110	
111	112	113	114	115	116	117	118	119	120	

n=100

```
for(i=2 ;i<=n;i++){
    if(che(i) 2
    j =
}
```

T.C O(n)

2 to n-1

a=b=√n

a×b

a<b

a<√n

b<√n

a×a=n

b×b=n

a=√n

2.3755

int divisor

```
while (div <=n/2
    if n%div==0{
        return false
    }
    divit
}
```

n=√n

sieve of Eratosthenes

										Prime numbers
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
	101	102	103	104	105	106	107	108	109	110
	111	112	113	114	115	116	117	118	119	120

Prime Sieve

n=121

n=1024

$\frac{N}{2}+\frac{N}{3}+\frac{N}{5}+\frac{N}{7}+\frac{N}{11}+\frac{N}{13}-$

$N(\frac{1}{2}+\frac{1}{3}+\frac{1}{5}+\frac{1}{7}\cdots)$

$N(\log_2 \log_2 N)$

$N \times \log_2 \times \log_2 N$ ~ O(n)

$\log_2 \log_2 10$

1

$\log_2 10 \log_2 \log_2 10$

$\log_2 10 = 3.$

$\log_2 \frac{10}{2}$

```
public static int Prime_Sieve(int n) {
    int[] prime = new int[n];
    prime[0] = 1;// not prime
    prime[1] = 1;// not prime;
    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;
            }
        }
    }
}
```

n=90

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