30 August 2023 22:02

Time and Space Complexity

```
for(i=1; i<=n; i++)
{
    print();
}
```

1....n

→ O(n)

1.... n-1

```
for(i=1; i<n; i++)
{ print();
}
```

→ O(n-1)

n= 10^6 ⟹ O(n)

n= 10^5 →

O(n+1) → O(n)

```
if (x%2==0)
    print("even");
```

→ O(1)

constant

```
for(i=1; i<=n; i=i+2)
{
```

1, 3, 5, 7.... n

{

}

$\rightarrow O(n/2)$

$\rightarrow$ $\llcorner O(n)$

$n = \underline{\underline{10^6}}$

$\frac{n}{2} = 5 \text{ Lakh} = \underline{\underline{5 \times 10^5}}$

```
for (i = 2; i <= n; i += 2)
{
    print();
}
```

2, 4, 6...

$\rightarrow O(n/2) \rightarrow O(n)$

i <= sqrt(n)

```
for (i = 1; i*i <= n; i++)
{
    print();
}
```

$\rightarrow O(\sqrt{n})$

$i^2 <= n$

$i <= \sqrt{n}$

$O(\sqrt{n})$

$n = 100 \Leftarrow$

$i = 1$ $1 \times 1 <= 100$

$i = 2$ $2 \times 2 <= 100$

$\vdots$ $\vdots$

$i = 9$ $9 \times 9 <= 100$

$i = 10$ $10 \times 10 <= 100$

$i = 11$ $11 \times 11 <= 100$

10 times

$O(\sqrt{n})$

$n = 10^6$ $\sqrt{n} \rightarrow 10^3$

10 Lakh 1000 = 1 thousand

```
for (i=1; i x i x i <= n; i++)
{
    print();
}
```

$O(\sqrt[3]{n})$

Cube root of n

```
for (i=1; i<=n; i++)
{
    break;
}
```

$\rightarrow O(1)$

```
for ( i=1 ; i<=n ; i=i x 2 )
{
  print();
}
```

$\rightarrow O(n)$

$n = 70$

$i = 1, 2, 4, 8, 16, 32, 64, 128$

$2^0 \quad 2^1 \quad 2^2$

$i = 1, 2, 4, 8, 16 \ldots\ldots n$

$\rightarrow 2^0 \quad 2^1 \quad 2^2 \quad \ldots\ldots \quad 2^k$

$\Rightarrow O(k+1)$

$\Rightarrow$

$$n = 2^k$$

$$2^k = n$$

take $\log_2$ on both sides

$$\log_a b^c = c \times \log_a b \qquad \log_2 2^k = \log_2 n$$

$\log_a b = (x) \log_a b$

$k \times \underline{\log_2 2} = \log_2 n$

$\log_a a = \underline{1}$

$k \times 1 = \log_2 n$

$k = \log_2 n$

$= O(k+1)$

$= O(k)$

$\Rightarrow O(\log_2 n)$

```
for (i=1; i<=n; i=i×3)
{
    print();
}
```

$\Rightarrow O(\log_3 n)$

```
for (i=n; i>=1; i=i/2)
{
    print();
}
```

$\Rightarrow$ 64, 32, 16, 8, 4, 2, 1

$\Rightarrow$ n, $n/2$, $n/4$, ... 16, 8, 4, 2, $\boxed{1}$

$\dfrac{n}{2^0}$ $\dfrac{n}{2^1}$ $\dfrac{n}{2^2}$ .... $\boxed{\dfrac{n}{2^k}}$ $\Rightarrow O(k+1)$

$\downarrow$

$O(k)$

$$\frac{n}{2^k} = 1$$

$$n = 2^k$$

take $\lg_2$

$$\lg_2 2^k = \lg_2 n$$

$$k \times \underline{\lg_2 2} = \lg_2 n$$

$$k = \lg_2 n$$

$\rightarrow O(k)$

$\rightarrow O(\lg_2 n)$

```
for (i=n; i>=1; i=i/x)
{
    print();
}
```

$O(\lg_x n)$

```
for (i=1; i<=n; i++)
{
    for (j=1; j<=n; j++)
    {
        print();
    }
}
```

i=1   i=2   i=3   ....   i=n

ntimes   ntimes   ntimes   ...   ntimes

$3+3+3=9=3^2$

$4+4+4+4=16=4^2$

$\underbrace{n+n+\dots n}_{\text{ntimes}} = n^2$

```
for (i=1; i<=n; i++)
{
    for (j=1; j<=i; j++)
    {
        ...
```

j1   j<=3

{
print();
}
}

i=1 i=2 i=3 i=4 ... i=n

1 times 2 times 3 times 4 times .... n times

$\Rightarrow 1+2+3 \ldots n$

$\Rightarrow$ Sum of n natural nos

$= \frac{n(n+1)}{2}$

$O(n)$

$O(\sqrt{n})$

$O(n/2)$

$O(n^2)$

$x^6 + 8x^5 + 10x^4 + 3x^2 + 4x + 7 = 0$

↳ degree = 6

$= \frac{n(n+1)}{2} = \frac{n^2+n}{2}$

$\Rightarrow O\left(\frac{n^2}{2} + \frac{n}{2}\right)$

$n = 10^3$

$\Rightarrow O(n^2 + n)$

$10^6 + 10^3$

$= O(n^2) + O(n)$

$\Rightarrow O(n^2)$

```
for( i=1; i<=n; i++)
{
  for( j=1; j<i; j++)
  {
    print();
  }
}
```

i=1   i=2   i=3 ...... i=n

0 times   1 time   2 times   n-1

$\Rightarrow 0+1+2+\ldots n-1$

$O(n^2)$

$1+2+3\ldots P \Rightarrow \frac{P(P+1)}{2}$

$$\frac{(n-1)(n-1+1)}{2} \Rightarrow \frac{(n-1)(n)}{2}$$

```
for( i=1; i<=n; i++)
{
  for( j=1; j<=n; j=j*2)
  {
```

```
{
  print();
  }
}
```

$i=1$ $i=2$ $i=3$ .... $i=n$

$\lg_2^n$ $\lg_2^n$ $\lg_2^n$ ... $\lg_2^n$

$$\Rightarrow \underbrace{\lg_2^n + \lg_2^n + \dots \lg_2^n}_{n\text{times}}$$

$2+2+2 = 6$
$= 3\times2$

$4+4+4 = 12$
$= 3\times4$

$$\Rightarrow n\lg_2^n$$

```
for( i=1; i<=n ;i++)
{
  for(j=n; j>=1; j=j/2)
  {
    print();
  }
}
```

$i=1$ $i=2$ .... $i=n$

$\lg_2 n$ $\lg_2 n$ $\lg_2^n$

$$\Rightarrow n\lg n$$

```
for (i=1; i<=n; i++)
{
    for(j=1; j<=n; j++)
    {
        if (j%2==1)
        {  break;
        }
    }
}
```

$i=1$ $\quad i=2$ $\quad i=3$ $\quad \dots \quad i=n$

1 time $\quad$ 1 time $\quad$ 1 time $\quad$ 1 time

$= \underbrace{1+1+\dots}_{n\ times} 1 \Rightarrow n\ times$

$O(n)$

```
for(i=1; i<=n; i++)
{
    for(j=1; j<=n; j++)      j=2
    {
        if (j%2==0)
        {  break;
```

$i=n$

} break;

}

}

}

$i=1$ $i=2$ . . . . . $i=n$

2 times   2 times   2 times

$$\Rightarrow \underbrace{2+2\ldots\ldots}_{n\text{ times}}\ 2$$

$$= 2n$$

$$O(2n)$$

$$\Rightarrow O(n)$$

Sum=0

for (i=1; sum<=n; i++)

{

Sum=sum+i;

}

1+2   1 ↑

$i=1$ $i=2$ $i=3$ $i=4$

sum=0   sum=1   sum=1+2   . . . .   sum=1+2+3

Sum <= n

1+2 . . . . p <= n

$1+2 \cdots p <= n$

$$\frac{p(p+1)}{2} = n$$

$$p^2 + p = 2n$$

$$p^2 + p - 2n = 0$$

$$ap^2 + ap - 2n = 0$$

Q. for ($i = 0$; $i < N$; i++) $\rightarrow O(N)$

$0 \ldots N-1$

{

a = a + rand();

}

for ($j = 0$; $j < m$; j++) $\rightarrow O(m)$

$0 \ldots m-1$

{

b = b + rand();

}

$TC \Rightarrow O(N+m)$

$SC: O(1)$

```
for (i=0; i<N; i++)
{
    for( j=N; j>i ;j--)
    {
        a= a+i+j;
    }
}
```

j=N ; j>N-1

i=0 i=1 i=2 ... i=N-1

N times N-1 times N-2 ... 1

$$N + (N-1) + \dots 1$$

$$\Rightarrow 1+2 \dots \dots (N-1) + N$$

$$= \frac{N(N+1)}{2} \rightarrow O(N^2)$$

```
for( i=n/2 ; i<=n ;i++)
{
    for (j=2 ;j<=n ;j=j*2)
    {
        k =k+n/2;
    }
}
```

i=n/2 i= n/2 +1 .... i=n

log n

}

$i = n/2 \quad i = n/2 + 1 \quad \ldots$

$\log_2 n \quad \log_2 n \quad \ldots \quad \log_2 n$

$$\Rightarrow \left(\frac{n}{2}\right) \log_2 n$$

$$\Rightarrow n \log_2 n$$

```
for ( i=0; i<n; i++)
{
    i = i x k;
}
```

$$\Longrightarrow O(\log_k n)$$

```
value = 0;
for( i=0; i<n; i++)
{
    for (j=0 ; j<i ; j++)
    {
        value += 1;
    }
}
```

j=0 ; j<2

$i=0 \quad i=1 \quad i=2 \quad \ldots \quad i=n-1$

0 times 1 time 2 times . . . n-1 times

$\Rightarrow 0 + 1 + 2 \ldots n-1$

$\Rightarrow 1 + 2 + 3 \ldots P$

$= \frac{P(P+1)}{2}$

$$\frac{(n-1)(n-1+1)}{2}$$

$$\Rightarrow \frac{(n-1)(n)}{2}$$

$$\Rightarrow O((n)(n-1))$$