07-09 Algorithmic Analysis - 2

07 September 2022 20:25

Rules:-

1. Ignore all constants:-

$O(5n) \Rightarrow O(n)$

n: 10,00,000 (3.12ms)

5n = 50,00,000 (3.89ms)

Q

```
for(i=0; i<n; i++) {
    Print(i)
}
for(j=0; j<n/2; j++) {

}
```

+ Steps!!!

TC:- O(   )

```
n = 8;
for(i=0;i<n;i++){
    console.log("i",i);
} //n
for(j=0;j<n/2;j++){
    console.log("j",j);
} //n/2
// O(n)+O(n) => O(n)
```

O(n)

O(n/2)

n comp. steps

$O(n) + O(n \div 2)$

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

$\approx O(n)$

Q

```
for(i=0; i<n; i++)
{
    for(j=0; j<n; j++) {
        Console.log(i,j) 1
    }
}
```

Steps y

Steps x

Ans:- ~~x+y~~ xy

n = 3

n = 5

5 < 5

i=0 => 3

i=1 => 3

i=2 => 3

5 < 5

n=5

i=0	j	i=1	j	i=2	i=3	i=4
5	j=0, j=1, j=2, j=3, j=4	5	j=0, j=1, j=2, j=3, j=4	5	5	5

25

n=5 => Steps taken 25

n=3 => Steps => 9

Worst case Time complexity => $O(n^2)$

Worst case — $O(n^2)$

Case Time Complexity

Imp

A: 20 B: 50 C: 100

$O(1) \rightarrow$ Constant Complexity

$O(n) \rightarrow$ Linear complexity

$O(n^2) \rightarrow$ Quadratic complexity

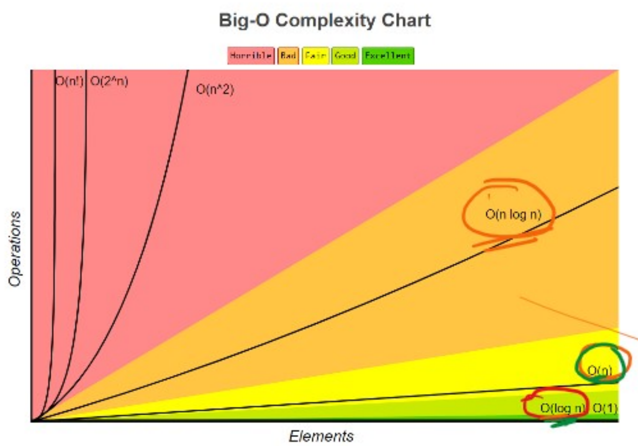
$O(\log n) \Rightarrow$ logarithmic complexity

$$O(n^2) + O(3n) + O(1)$$

$$O(n^2) + O(n) + O(1)$$

Worst case

$$\approx O(n^2)$$



$\log 2 \Rightarrow 0.3010$

```
> n = 5; count = 0;
  for(i=0;i<n;i++){
     for(j=0;j<n;j++){
        count++

     }
  }
  console.log(count)
  25
```

$O(n)$ $O(n)$

$O(n*n)$

$O(n^2)$

```
for(i=0; i<5; i++)
{
    for(j=0; j<5; j++)
    {

    }
}
```

```
i=0  for(i=0; i<5; i++) i=0
     {
         for(j=0; j<5; j++) j=5
         {

         }
     }
```

5

$O(n)$

}   }
+
i=1
for(i=1 ; i<5 ; i++)   O(n)
{   *
    for(j=0;j<5;j++)   O(n)
    {
    }
}
+

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

O(n)

O(n)+O(1)

O(n²) → $O(n^2)$

TC:- $O(n^2)$

```
x = 5 + (15 x 20)   // O(3)
for(i=0; i<n; i++){
        Print(n)
}
            i<5
for(i=0 ; i<n ; i++)  {5
    for(j=0; j<n; j++){
        Print(i,j)
    }
}
```

O(n)

O(n)

O(n)

5n

$O(n^3)$

$O(3)+O(n)+O(n^2)$

$O(n^2)$

Q

```
if x>0
    // O(1)
else if  x<0
    // O(log n)
else
    // O(n^2)
```

x=5 // O(1)

x=-1 // O(log n)

x=0 // O(n²)

$\boxed{O(n^2)}$

Q

```
let a = n;
while (a > 0) {
    a = ParseInt(a/2)
}
```

n=16	n=20	n=25
a=8	a=10	a=12
a=4	a=5	a=6
a=2	a=2	a=3
a=1	a=1	a=1
a=0	a=0	a=0
8 steps => 5	5	

$\log n + 1$

5 $\log n - 1$

$\log_2 25$ → 5

$\log_2 n + 1$

$\log_2 16 \Rightarrow 4 \approx \boxed{\log_2 n}$

```
for(i=1 ; i<=n; i=i*2) {
    Print (i)
}
```

n=20

i=1, i=2, i=4, i=8, i=16 — 5 steps

i=32 XX

4.3

$\log_2 n$

n=16

i=1, i=2, i=4, i=8, i=16 — 5 steps

$\log n + 1$

n=36

i=1, i=2, i=4, i=8, i=16, i=32, ~~i=64~~

$\log 36$ → 6

$\log n$

```
for(i=0; i<n; i++)   // O(n)
{
    if (   ) {
        if (   ) {
            ==
        }                O(log n)
        --- O(n)
                         O(n) + O(log n)
    }                                        O(n)
    else {
        O(n)             O(n)
    }
}
```

O(n) + O(n)

O(2n) ≈ O(n)

O(n) * O(n) ≈ $O(n^2)$

```
for(i=0; i<n; i++)  // O(n)
{
    if(   ) {
        if(   ) {
            ---            O(logn)
            ---
        }
        --- O(n)                      O(n)
    }
    else {
        O(n logn)
    }
```

$O(n\log n)$

$O(n) * O(n \log n)$

$\{O(n^2 \log n)\} \to$ worst case

$\downarrow g(n)$

* Notations:-

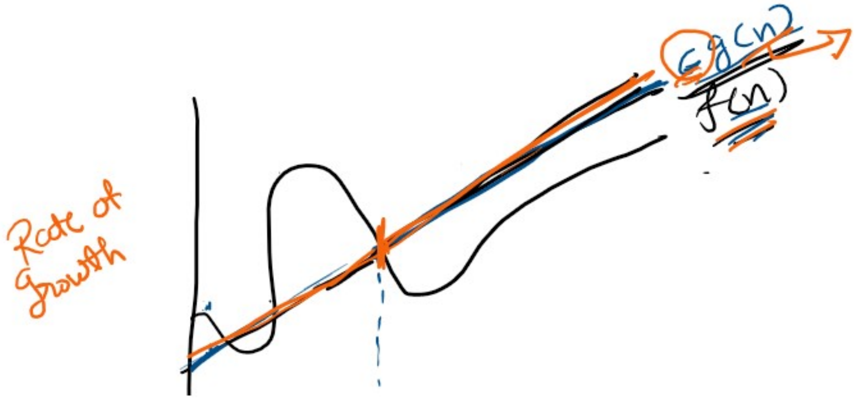
1. Big-oh notation $\to$ worst case $O(\square)$
2. Theta notation $\to$ avg case $\Theta(\ )$
3. Omega notation $\to$ best case $\Omega(\ )$

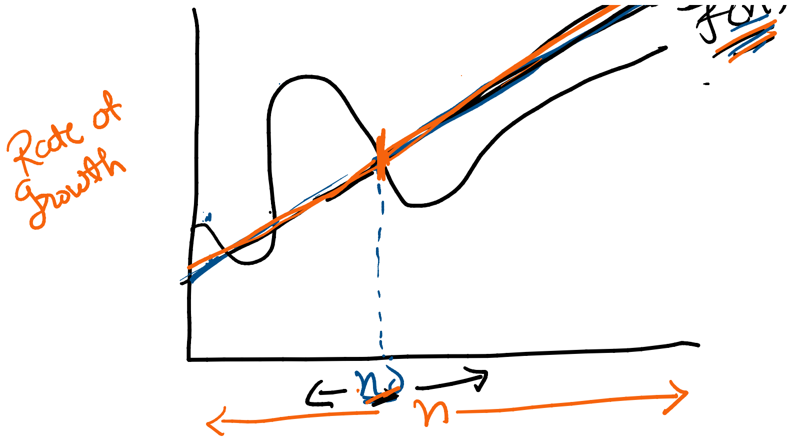
$\Rightarrow$ Big-oh notation $\to$ give you tight upper bound of algo.

$$\boxed{f(n) = O(g(n))}$$

at larger values of n, the tight upper bound of f(n) is g(n).

$O(1) \to O(n)$





$$f(n) = n^4 + 100n^2 + 10n + 50$$

$$g(n) = n^4$$

$O(g(n))$:- { $f(n)$:- there exist positive constants c & n_0 such that

$$0 \leq f(n) \leq g(n)$$

for all $n \geq n_0$

Q $f(n) = 3n + 8$

$3n+8 \leq g(n)$ ⇒ for all $n \geq n_0$

c, $g(n)$ & n_0.

$$\boxed{3n+8 \leq 4n}$$

$n=8$ $n=7$ f $n=9$ t

$f(n) \leq c\, g(n)$;

↳ $n \geq n_0$

$c = 4$ $n_0 = 8$

$g(n) = n$

8

n0

$3n+8 \le 4n$ $\quad n=1$

$11 \le 4$ ✗ $\quad n=7$ → $29 \le 28$ $\quad n=8$ $\quad \boxed{32 \le 32}$

Something

tight

5sec

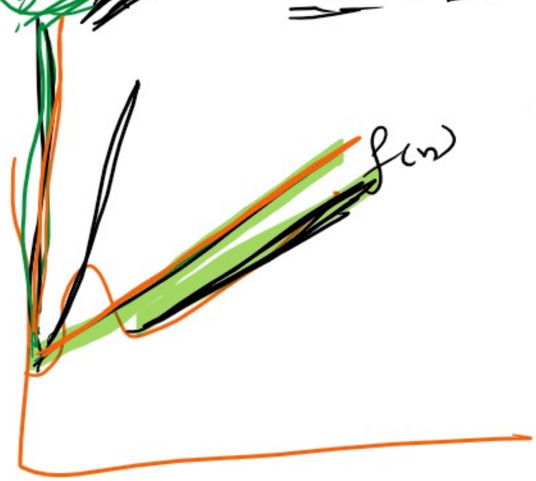
g(n) → upper bound

MRP => 90

120

50

1000



$3n+8 \le 10n$ } $\quad n_0 = 2$

$n \ge 2$

$n < 2$ ✗

$n = 1$

$11 \le 10$ $\quad 13$ ✗

$n = 2$

$14 \le 20$ ✓

90%

$\boxed{3n+8 \le 4n}$ ✓

$\boxed{3n + 8 \le 4n}$ ✓

+1

Aptitude   Linkedin   {}   10 (2/3)

Q $f(n) = n^2 + 1$
find out $g(n)$, C, n_0

$f(n) \le c\,g(n)$; $n \ge n_0$

$n^2 + 1 \le 2n^2$

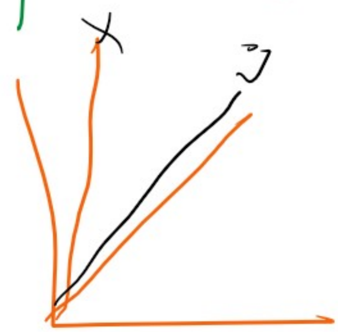
$C = 2$ - $n_0 = 1$

$g(n) = n^2$

n^2+1 } $O(n^2)$
$O(g(n))$

Q $100n + 5$
find $g(n)$, n_0, c

$g(n) = n$



$f(n) \le c\,g(n)$; $n \ge n_0$

$100n + 5 \le 106n$

$n = 1$ $105 \le 106$
$n = 2$ $205 \le 212$

$C = 106$
$g(n) = n$
$n_0 = 1$

$100n + 5 \le 105n$ ✓

$\boxed{100n+5} < 101n$

$n = 1$
$105 \le 100$ ✗✗

$n = 1$ $105 \le 101$ ✗

$n = 1$
$105 \le 101$ ✗✗

$C = 101$
$g(n) = n$

$n = 5$ $\boxed{505 \le 505}$

$n=1$

$105 \leq 101$ XX

$n=2$

$C=101$

$f(n)=n$

$n_0=5$

$n=5$

$\boxed{505 \leq 505}$

$O(1) \rightarrow$ Constant

$O(n) \rightarrow$ linear

$O(\log n) \rightarrow$ logirithmic

$O(n^2) \rightarrow$ Quadratic

$O(n^3) \rightarrow$ cubic

$O(n!) \Rightarrow$ factorial

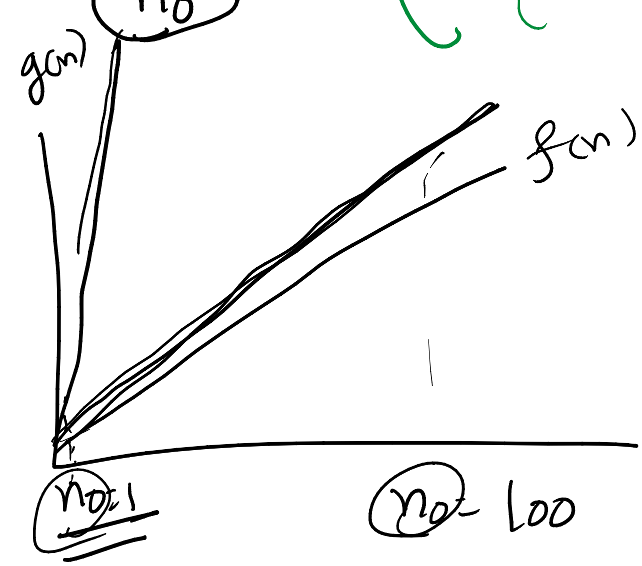
$O(2^n) \Rightarrow$ Power complexity

$f(n) \Rightarrow$

$O(\boxed{n \log n})$

Point

n_0



Javascript

{React js} XX

Nodejs }}

APIs
Axios
Network

dbs

abcd