

Time Complexity:-

Recursion:-

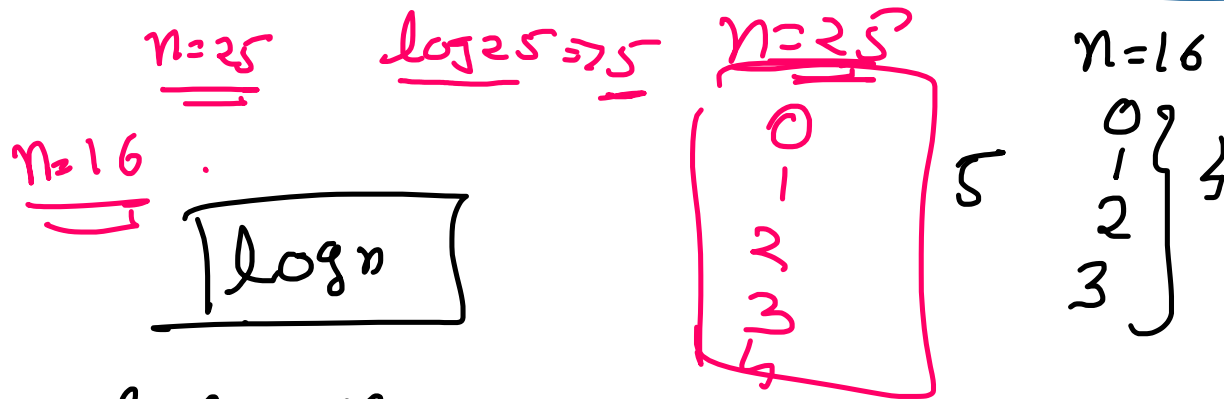
Bubble  
Insertion  
Selection } (TC)

Q function f1(n) { O(n)  
     for(i=0; i<n; i++)  
     {  
         console.log(i);  
     }  
 }

n=5  
 i=0  
 i=1  
 i=2  
 i=3  
 i=4 } 5 }  
 n=7 } 7 } O(n)

Q function f2(n) {  
     for(i=0; i< Math.log(n); i++)  
     {  
         Math.log(25) 5  
         log 16 => 4  
     }  
 }

$\underbrace{\text{console.log}(i)}_{\text{O}(1)} = \boxed{\underbrace{\text{O}(n^2)}_{\text{O}(n \log n)}}$



Q function f3(n, m) {  
 for (i = 0; i < n; i++) {  
 console.log(i);  
 }  
 for (j = 0; j < m; j++) {  
 console.log(j);  
 }  
}

$\underbrace{\text{O}(1)}_{\text{times}}$

$\underbrace{+}_{\text{m times}}$

$\text{O}(n+m)$

$\text{O}(n^2 + m)$   
 $\text{O}(n^2)$

Q function f4(n, m) {  
 for (i = 0; i < n; i++) {  
 {

$\text{O}(n+m) \Rightarrow \text{O}(n^2)$

$n$

$m$

$3$

$3$

$\{$

for( $j=0; j < m; j++$ ) {

  console.log( $i, j$ )

$\}$

$n \cdot m$

$O(n^2 m)$

2 example

---

$\{$

$n=3$

$m=2$

$\}$

6 times { $n \cdot m$ }

$i=0$

$j=0$

$j=1$

$i=1$

$j=0$

$j=1$

$i=2$

$j=0$

$j=1$

Q5 function f5(n)

```

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

```

$n=5$

$i=0$

$j=i+1$

$0+1=1$

$i=1$

$j=2$

$i=2$

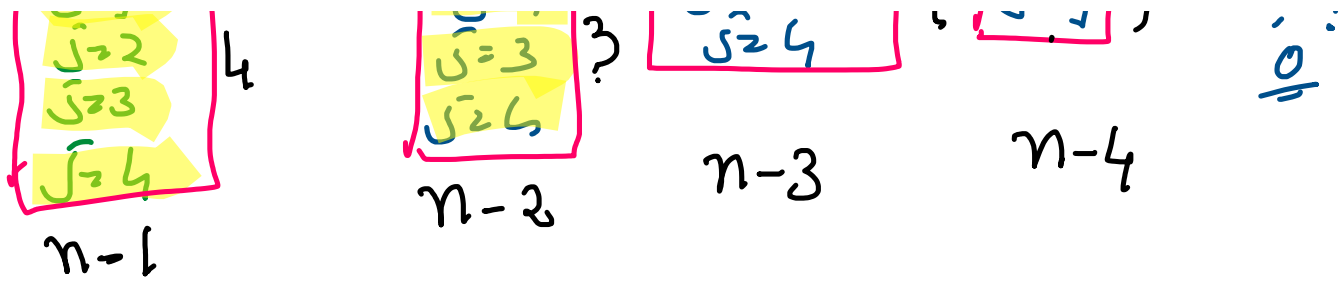
$j=3$

$i=3$

$j=4$

$i=4$

$j=5$



$$\begin{aligned}
 & \left. \begin{array}{l} n-1 \\ n-2 \\ n-3 \\ \vdots \\ 0 \end{array} \right\} (n-1) + (n-2) + (n-3) + \dots + 0 \\
 & \frac{n(n+1)}{2} \approx \boxed{n^2} \\
 & \quad \quad \quad \text{O}(n^2)
 \end{aligned}$$

Q //

function  $f(n)$  {

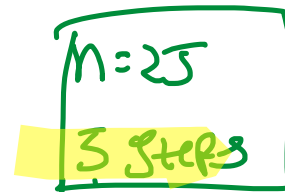
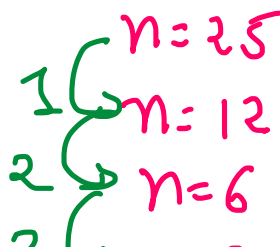
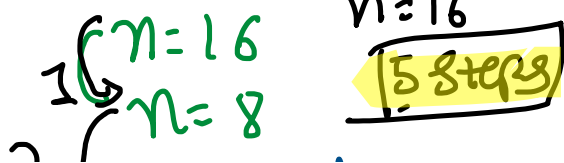
while ( $n > 0$ ) {

console.log( $n$ );

$n = n/2$ ;

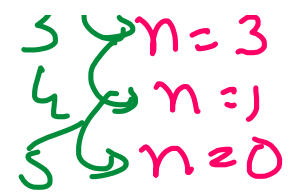
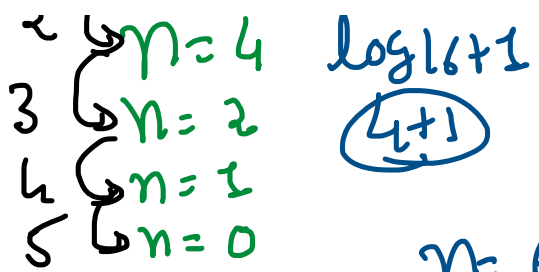
}

}

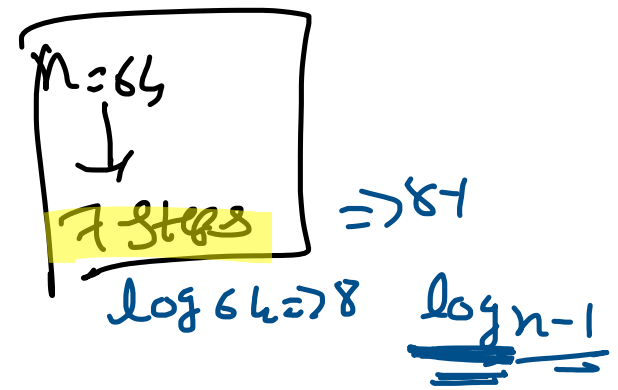
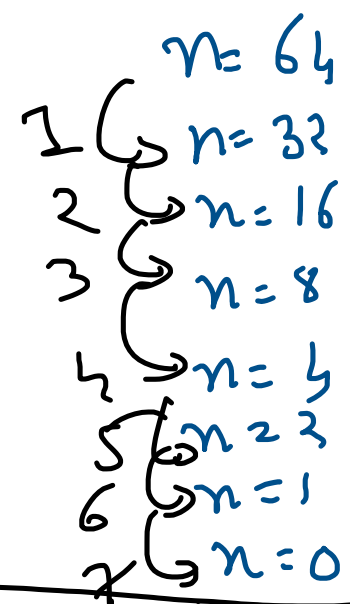
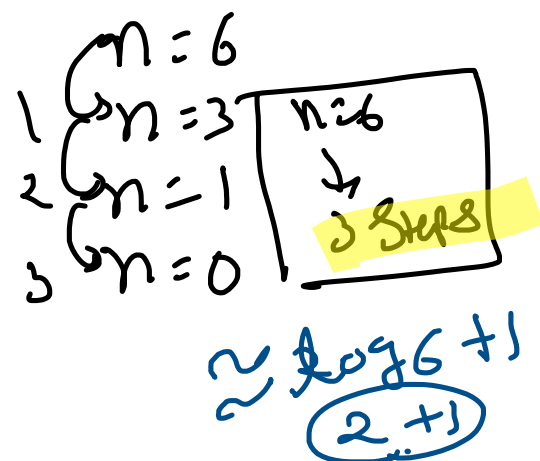


$\log 25$   
 $\log n$

$\log n + 1$   
 $\log n$   
 $\log n + 1$



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



$\log m^n$   
 $\hookrightarrow n \log m$   
 $\log m + \log n$   
 $\hookrightarrow \log mn$

**Q** function  $f(n)$  {  
 - let  $i=1$ ;  
 - let  $pow=1$ ;  
 while ( $i < n$ ) {  
    $i = pow$ ;  
    $pow = 2$ ;  
 }  
 }

①  $i=1$     ②  $i=1$

$n=8$   
 ①  $i=1$   
 $pow=2$   
 ②  $i=2$   
 $pow=4$   
 ③  $i=4$   
 $pow=8$   
 ④  $i=8$   
 $pow=16$

~~⑤~~

⑤

$n=16$      $pow=1$  }  $pow=2$      $i=4$      $i=8$      $i=16$      $10^{80}$   
            $(2) \quad i=2$      $pow=8$      $pow=16$      $pow=32$   
            $pow=4$

$n=8 \rightarrow 4 \quad \} \quad \log 8 + 1 \Rightarrow 3+1$   
 $\log n + 1$

$n=16 \rightarrow 5 \quad \} \quad \log 16 + 1$   
 $\log n + 1$

$\log 16 + 1$   
 $\downarrow$   
 $4$

$\log n + 1$

$O(\log n + 1)$   
 $O(\log n)$

$Q \equiv$

function  $f(n)$  {

for ( $i=0$ ;  $i < n$ ;  $i++$ ) {

for ( $j=0$ ;  $j < \text{math.log}(n)$ ;  $j++$ ) {

console.log(i, j)

$n$   
 $\log n$   
 $3$

$O(n \log n)$

4

function f(n) {

for (let i = 0; i < n; i++) {  
 console.log(i);  
}

$O(n)^n$

~~$O(n^2)$~~   
 $O(n^2)$

for (let j = 0; j < n; j++) {

$O(n^2)$

for (let i = 0; i < n; i++) {  
 console.log(i);  
}

$O(n^2)$

$m, n$

$n, n^2, n^3$   
 $n=2$

Q

let ans = 0;

for (let i = 1; i < n; i++) {

$n-1$

for (let j = n; j > 1; j--) {  
 ans += i;  
}

$n-1$

$(n-1) * (n-1)$   
 $n^2$

$O(n^2)$

2. for (let i = n; i > 0; i = i/2)

```

// {
//   for (let j = 0; j < i; j++) {
//     console.log(i+j);
//   }
// }

```

$$i = n$$

$$i = n/2$$

$$i = n/4$$

$$j = n$$

$$j = (0, n/2)$$

$$j = (0, n/4)$$

↳ operation

↳  $n/2$  operation

↳  $n/4$  operation

$$\{ n + n/2 + n/4 + n/8 + \dots \}$$

$$S = \frac{a(1-r^n)}{1-r}$$

$$a = n$$

$$r = 1/2 \rightarrow \text{diff}$$

$$\text{no. of terms} = \log n$$

$$= \frac{n \left( 1 - \frac{1}{2}^{\log n} \right)}{1 - \frac{1}{2}}$$

$$= 2n \left( 1 - \frac{1}{2}^{\log n} \right)$$

first term  
[AP, GP, HP]



$$= 2n \left( 1 - \frac{1}{2^{\log_2 n}} \right)$$

$$= 2n \left( 1 - \frac{1}{n} \right)$$

$$y = 2^{\log_2 n}$$

$$\log y = \log 2^{\log n}$$

$$= \log n \log 2$$

$$\log_2 y = \log_2 n$$

$$y = n$$

$$= 2n \left( \frac{n-1}{n} \right)$$

$$a = \log 2$$

$$b = \log n$$

$$\log a = 0$$

$$b \log a$$

$$\log b$$

$$2(n-1)$$

$$= 2n - 2$$

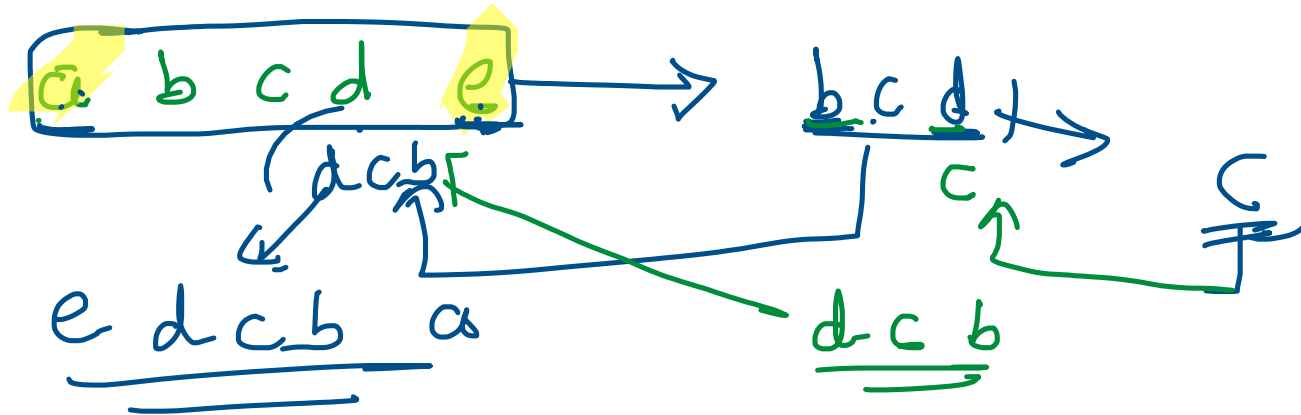
$$= O(n)$$

Q Reverse a string using recursion

IP "abcde" → OP "edcba"

Q1 abcde ① ② ③

Hint:-



①  
a  
a

Q2 [1 1 0 1 1 0]  $\Rightarrow 3$

max count: - 0 2 3

Count: - 1 2 0 1 2 0

max count > count 0 < 2

max count < count 2 < 3

[1 0 1 1 0 0 1 1]

count > max count

Count 0 1 0 1 2 3 0 1 2  
Max Count - 0 1 3

Count - reset

$0 < 1$        $3 > 2$   
 $1 < 3$