Example 1:

Time

Space ✓

```
swap (a,b)
{
    temp = a;   → 1
     a = b;     → 1
     b = temp;  → 1
}
```

$$\boxed{f(n) = 3 \Rightarrow O(1)}$$

a → 1

b → 1

temp → 1

$$\boxed{S(n) = 3 \text{ bytes} \quad O(1)}$$

space

Complexity

$x = 5*a + 6*b \rightarrow 1$

$x = 5*a + 6*b \quad 1$

$x = 5*a + 6*b \quad 1$

$x = 5*a + 6*b \quad 1$

$f(n) = \longrightarrow 4 \Rightarrow \boxed{O(1)}$

↓

constant

Constant

Time Complexity

1. Constant Complexity

Worst Case ⇒ O (Constant) ⇒ $O(1)$

Example 2

A =

0	1	2	3	4	5
8	3	9	7	2	4

$n=6$ $i=0,1,2,3,4,5$

```
Sum (A, n)                              Time
{
    s=0;                    ------->    1
    for (i=0; i<n; i++)     ------->    n+1
          n+1
    {
        s = s + A[i];       ------->    n
    }
}
n
```

Space

$A = n$

$n = 1$

$S = 1$

$i = 1$

$S(n) = n+3$

return S; $\longrightarrow$ 1 $\boxed{O(n)}$

}

$f(n) = 1 + n + 1 + n + 1$

$= \underbrace{2n}_{n} + \underbrace{3}_{const}$

$= \boxed{O(n)}$

↓

Linear Time Complexity

2. Linear Complexity $= O(n)$

n=3 i=3, j=3

9 ← 3×3

Example 3

Time

Space

Add (A, B, n) $\rightarrow 3$

$n = 3$

$n \times n$

n^2

$A \rightarrow n^2$

{ for(i=0; i<n; i++) $\longrightarrow$ n+1 ✓

m

$\rightarrow n^2$

```
for(i=0; i<n; i++)
{
    for(j=0; j<n; j++)          → n × (n+1)
    {
        C[i,j]=A[i,j]+B[i,j]    → n (n)
    }
}
}
```

$\underset{m}{n} \times (n+1)$ ✓

$\underset{m}{n}\,(\overset{n}{n})$ ✓

B → n

C → n^2

n → 1

i → 1

j → 1

$f(n) = 3n^2 + 3$

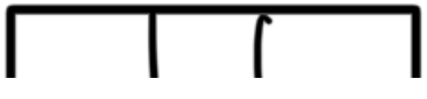
$\boxed{O(n^2)}$

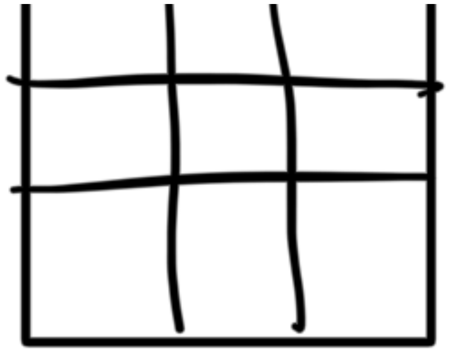
$f(n) = n+1 + n^2+n + n^2$

$= 2n^2 + 2n + 1$

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

3. Quadratic Complexity ⇒ $O(n^2)$





$A = B = C = n^2$

$n \times n \Rightarrow n^2$

$n = 3$ 3×3

$n = 5$ 5×5

$m \times n = \underline{3 \times 2} = \textcircled{6}$

$\frac{n \times n}{m \times m} = O(n \times n)$

$O(n^2)$

Example

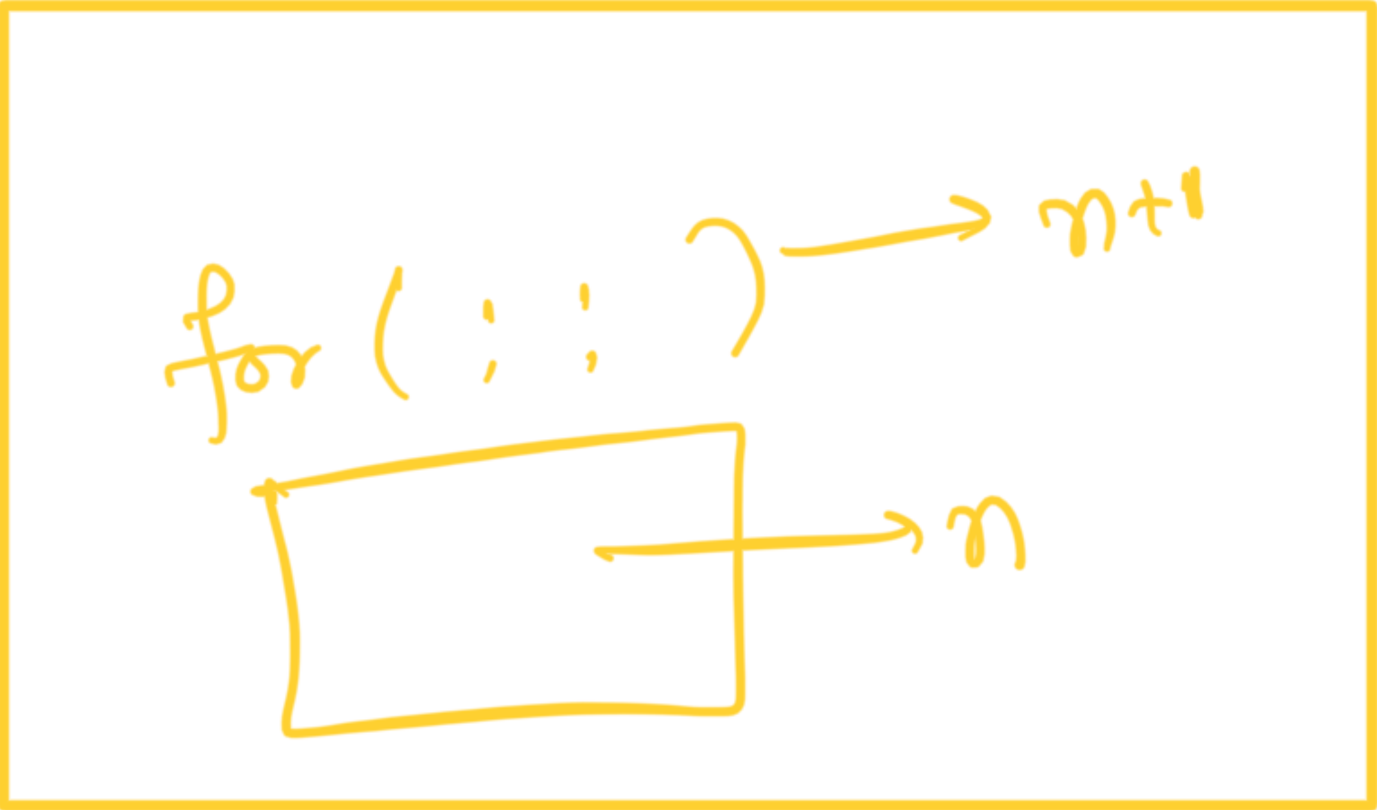
```
for (i=0; i<m; i++)  ⟶  m+1
{
    for (j=0; i<n; j++)  ⟶  m(n+1)
    {
        c[i,j] = A[i,j] + B[i,j];  ⟶  m (n)
    }
}
```

}

$m \Rightarrow$ 1 word } $\dfrac{n \times n}{m \times m}$

$n \Rightarrow$ 1 word

$$
\begin{aligned}
S(n) &= m+1 + mn + m + mn \\
&= 2mn + 2m + 1 \\
&= 2n^2 + 2n + 1 \\
&= O(n^2)
\end{aligned}
$$



Example 4 :

```
Multiply (A, B, n)

for (i=0; i<n; i++) ————— n+1
{ for (j=0; j<n; j++) ————→ n x (n+1)
   {      c[i,j]=0; ————→ n x n *1

          for (k=0; k<n; k++) ——→ n x n x (n+1)
          {  c[i,j]= c[i,j] + A[i,k] * B[k,j]; → n x n x n
          }
   }
}
```

$$f(n)= n+1+n^2+n+n^2+n^3+n^2+n^3$$

$$= 2n^3+3n^2+2n+1$$

$$= O(n^3)$$

space →

A	n^2
B	n^2
C	n^2
n	1
i	1
j	1
k	1

$$S(n) = 3n^2 + 4$$

$$O(n^2)$$

$f(n)$ ⇒ Time Complexity

① for (i=0; i<n; i++) → $(n+1)$

{

{

st $\longrightarrow$ n

}

$\boxed{O(n)}$

② for(i=n; i >0; i--) $\longrightarrow$ $(n+1)$

{

st $\longrightarrow$ n $\boxed{O(n)}$

}

③ for (i=1; i < $\overset{10}{n}$; i+2) $\longrightarrow$ $\frac{n+1}{2}$

{

— st $\longrightarrow$ $\frac{n}{2}$

}

$f(n) = \frac{n}{2}$

$\boxed{O(n)}$

④
```
for(i=1; i<n; i = i+20)
  {
        —— st
  }
```

$f(n) = \frac{n}{20}$

$\boxed{O(n)}$

⑤
```
for (i=0; i<n; i++)        — n+1
  { for(j=0;j<n; j++)      n×(n+1)
      {
          stm—      n×n
      }
  }
```

$f(n) = n+1 + n^2 + n + n^2$

$= 2n^2 + 2n+1$

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

⑥
```
for (i=0; i<n; i++)
```

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

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

constant → Linear → Quadratic → Cubic → O^n → 2^n → n! → variety

Example -

1,2,n

```
for(i=0;i<n;,i++)
{  for(j=0;[j<i];j++)
```

i	j	times
0	0	×
1	0 ✓	✓

```
        {
            stm---
        }
}
```

	1 ×	
2	0 ✓ 1 ✓ 2 ×	2 ✓
3	0 1 2 3 ×	3
⋮		⋮
n	0 – ⋮ n-1 n ×	n

space = i — 1
j — 1
n — 1

$S(n) = 3 \Rightarrow O(1)$

$$1+2+3+ \cdots n = \boxed{\frac{n(n+1)}{2}}$$

Loop execution

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

ignor

$= O(n^2)$

Example

3 ~~2~~   * 15

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

$\sqrt{n}$

i	p
1	$0+1=1$
2	$0+1+2$ $1+2=3$
3	$0+1+2+3$ $3+3=6$
4	$0+1+2+3+4$ $6+4=10$
5	$0+1+2+3+4+5$ $10+5=15$
⋮	
k	$0+1+2+3+\cdots k$

sum of n no's

$n(n+1)$

Assume $\boxed{p > n \, (k)}$

$p > k$

$\therefore$

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

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

$\boxed{k^2 > n}$ $\boxed{k \Rightarrow \frac{k(k+1)}{2}}$

$\boxed{k > \sqrt{n}}$

$O(\sqrt{n})$ →

Example:

```
for(i=1; i<n; i=i*2)
{
    for( ______ i= i/2)

}
```

$\overset{2}{1}$ $*$

i
1
$1 \times 2 = 2$
$2 \times 2 = 2^2$
$2^2 \times 2 = 2^3$
$2^3 \times 2 = 2^4$
⋮

Assume $i > n$

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

$$i = 2^k$$

$$2^k >= n$$

$$2^k = n$$

$$k = \log_2 n \longrightarrow \boxed{O(\log_2 n)}$$

Important

$for(i=n; i>1; i=i/2) \longrightarrow O(\log_2 n)$

$for(i=1; i<n; i=i*2) \longrightarrow O(\log_2 n)$

$\longrightarrow O(\log n)$

for (i=1; i<n; i=i*3) $\longrightarrow O(\log_3 n)$

Types of Time functions –

$O(1) \longrightarrow$ Constant

$O(\log n) \longrightarrow$ Logarithmic

$O(n) \longrightarrow$ Linear

$O(n^2) \longrightarrow$ Quadratic

$O(n^3) \longrightarrow$ Cubic

$O(2^n) \longrightarrow$ Exponential

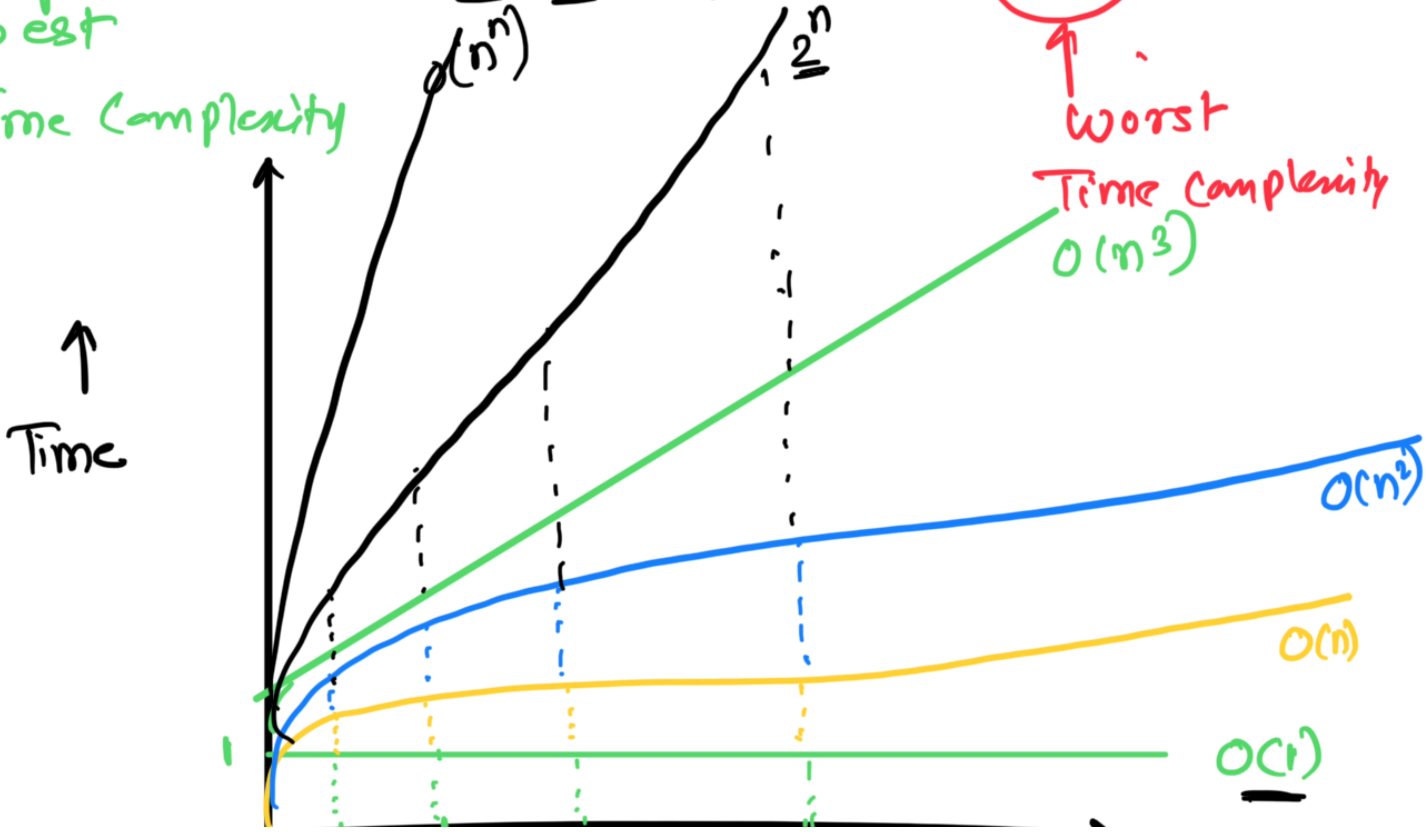
$O(3^n) \longrightarrow$

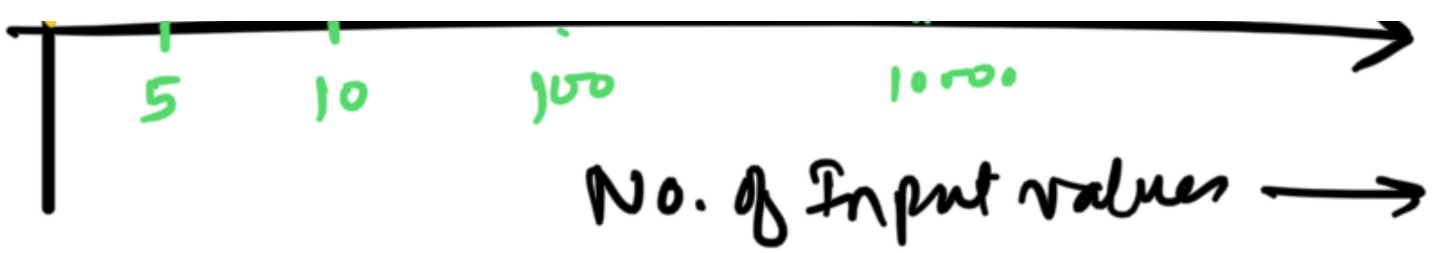
$O(n^n)$

$$1 < \log n < \sqrt{n} < n < n\log n < n^2 < n^3 < \cdots\cdots < 2^n < 3^n \cdots < n^n$$

Best Time Complexity

Worst Time Complexity





eg

A[n] → for (—— , ,) → O(n) Linear

A[2] → Logic → Constant

A[23]

A[11]

O(1) → Best

O(logn) – Good

O(n) – fair

O(nlogn) – Bad

[illegible] Worst as compared to O(1)

$O(n!)$ $O(c$ $)$, $O(n$ $)$ — Worst as compared to $O($ $)$

→ <u>Worst Case</u>

Asymptotic Notation

O → big-oh — Worst Case → <u>Upper bound</u>

Ω → big-omega — Best Case → Lower bound

Θ → theta — Average Case → Average bound