Time Complexity & Space Complexity

* Time complexity :
  - Big Omega (Best case)
  - Big Theta (Avg case)
  - Big-oh (Worst case)

* Things to Remember while find T:C

- Ignore constant
- Always consider highest values
- Infinity value

N	$f(n) = 3n + 3$	$<$	$g(n) = n^2 + 2$
1	6		3
2	9		6
3	12		11
4	15		18
5	18		27
6	21		38

smaller

Bigger

1. $2n^2 + 4n + 6$

$O(n^2)$

2. $5n^2 + 8n^2$

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

3. $\frac{n}{2} + n + n^2$

$O(n^2)$

4. $\frac{n}{100} + 200n$

$O(N)$

5. $100n$

$O(N)$

6. 100

$O(1)$

(7) n

(8) $n \log n$

(9) ~~10~~ $N \cdot \log N$

(10) $\frac{N \cdot \log N}{\cancel{100}}$

Increasing order ?

$N < N \cdot \log N$

```
for(i=1 ; i<=100 ; i++)
```
$O(1)$

```
for(i=1 ; i<=n ; i++)
```
$O(N)$

```
for(i=n ; i>=0 ; i=n/2)
```
$O(\frac{N}{2}) = O(N)$

```
for(i=1 ; i<=n ; i=i*2)
```
$O(N)$

for (i=1 ; i ≤ n ; i = i*2) $O(\log_2 N)$

$i = 1 = 2^0$

$i = 2 = 2^1$

$i = 4 = 2^2$

$i = 8 = 2^3$

$i = 16 = 2^4$

$i = 32 = 2^5$

$$2^K \leq N$$

$$\log 2^K \leq \log N$$

$$K \log 2 \leq \log N$$

$$K \leq \frac{\log N}{\log 2}$$

$$K \leq \log_2 N$$

for (i=1 ; i ≤ N ; i=i*3) $(\log_3 N)$

$i=1 = 3^0$

$i=3 = 3^1$

$i=9 = 3^2$

$i=27 = 3^3$

$\vdots$

$3^k \leq N$

$k \log 3 \leq \log N$

$k \leq \frac{\log N}{\log 3}$

$\boxed{k \leq \log_3 N}$

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

$O(n^2)$

```
for (i=1 ; i<=n ; i++)
{
    for (j=1 ; j<=n ; j=j*4)
    {
    }
}
```

$O(N \cdot \log_4 N)$

input size	required time complexity
$n \leq 10$	$O(n!)$
$n \leq 20$	$O(2^n)$
$n \leq 500$	$O(n^3)$
$n \leq 5000$	$O(n^2)$
$n \leq 10^6$	$O(n \log n)$ or $O(n)$
n is large	$O(1)$ or $O(\log n)$

S.NO	Big O Notation	Name
1.	O(1)	Constant Time Complexity
2.	O(log n)	Logarithmic Time Complexity
3.	O(n)	Linear Time complexity
4.	O(n log n)	Linearithmic Time Complexity
5.	O(n^2)	Quadratic Time Complexity
6.	O(n^3)	Cubic Time Complexity
7.	O(n^y)	Polynomial Time Complexity
8.	O(2^n)	Exponential Time Complexity
9.	O(n!)	Factorial Time Complexity

Algorithm (applied to Array)	**Time Complexity**		
	Best Cases	**Average Cases**	**Worst Cases**
Bubble sort	$O(n)$	$O(n^2)$	$O(n^2)$
Selection sort	$O(n^2)$	$O(n^2)$	$O(n^2)$
Insertion sort	$O(n)$	$O(n^2)$	$O(n^2)$
Shell sort	$O(n \log(n))$	$O(n \log^2(n))$	$O(n \log^2(n))$
Merge sort	$O(n \log(n))$	$O(n \log(n))$	$O(n \log(n))$
Quick sort	$O(n \log(n))$	$O(n \log(n))$	$O(n^2)$
Heap sort	$O(n \log(n))$	$O(n \log(n))$	$O(n \log(n))$
Counting sort	$O(n+k)$	$O(n+k)$	$O(n+k)$
Bucket sort	$O(n+k)$	$O(n+k)$	$O(n^2)$
Radix sort	$O(nk)$	$O(nk)$	$O(nk)$