• Polynomial grows faster than polylog. ex: $n = \omega(\log^{999} n)$, $n^{0.1} = \omega(\log^{99} n)$, $n \log^2 n = o(n^{1.01})$

• Limits: 1) $\lim_{n \to \infty} f(n)/g(n) \neq 0, \infty$ $f = \Theta(g)$

2) $\lim_{n \to \infty} f(n)/g(n) \neq \infty$ $f = O(g)$

3) $\lim_{n \to \infty} f(n)/g(n) \neq 0$ $f = \Omega(g)$

4) $\lim_{n \to \infty} f(n)/g(n) = 1$ $f \sim g$

5) $\lim_{n \to \infty} f(n)/g(n) = 0$ $f = o(g)$

6) $\lim_{n \to \infty} f(n)/g(n) = \infty$ $f = \omega(g)$

• Summation:

1) $\Sigma n \Rightarrow [n(n+1)/2] \Rightarrow n^2$

2) $2+4+8 \dots + n = 2^{(n+1)} - 2$.

3) Sum of n^2 terms $\Rightarrow [n(n+1)(2n+1)/6]$

4) sum of first n^3 terms $\Rightarrow \left[\frac{n(n+1)}{2}\right]^2$

5) sum of power of 2 terms $\Rightarrow 2^{n+1} - 1$
$[1+2+4+8 \dots n]$

• Master Theorem: $T(n) = a\,T(n/b) + O(n^d)$

$$T(n) = \begin{cases} O(n^d) & \text{if } d > \log_b a \\ O(n^d \log n) & \text{if } d = \log_b a \\ O(n^{\log_b a}) & \text{if } d < \log_b a. \end{cases}$$

• Binary search : $T(n) = T(n/2) + O(1)$

• Merge sort : $T(n) = 2T(n/2) + O(n)$ (Recursive

• Strassen : $T(n) = 7T(n/2) + n^2$.

• Merge sort : $O(a+b)$ [non Recursive]

• $\sum_{i=0}^{\infty} r^i = \frac{1}{1-r}$ for $|r| < 1$

$\sum_{i=0}^{\infty}$ sum $\Rightarrow \frac{a[r^n - 1]}{r-1}$ for $|r| > 1$

Imp Points

1) Selection sort → choose smallest element compare it with every element starting from 1st element than swap it with 1st ele. similarly keep sorting in that way.
Comp: $O(n^2)$.
ex: 1 2 3 1 5 4   2 1 3 2 5 4   3 1 2 3 5 4 ...

2) Change of base rule:

$$\log_a n = \Theta(\log_b n)$$

$$(\log_a b)(\log_b n) = \log_a n.$$

$$c \log_b n = \log_a n$$

so $\log_b n = \Theta(\log_a n)$

3) Asymptotic notation

$f(n) = O(g(n))$	f(n) doesnot grow faster than g(n)
$f(n) = o(g(n))$	f(n) grows strictly slower than g(n)
$f(n) = \Omega(g(n))$	" " atleast as fast as g(n)
$f(n) = \omega(g(n))$	f(n) grows strictly faster than g(n)
$f(n) = \Theta(g(n))$	" " at same rate.

	Ω Best	Θ Avg	O Worst
Selection :-	n^2	n^2	n^2
Bubble :-	n	n^2	n^2
Insertion :-	n	n^2	n^2
Heap :-	$n\log n$	$n\log n$	$n\log n$
Quick :-	$n\log n$	$n\log n$	n^2
Merge :-	$n\log n$	$n\log n$	$n\log n$
Bucket :-	$n+k$	$n+k$	n^2
Radix :-	nk	nk	nk
Count :-	$n+k$	$n+k$	$n+k$
Shell :-	$n\log n$	$n\log n$	n^2

Linear :-	1	n	n
Binary :-	1	$\log n$	$\log n$

⇒ always use lim

↳ if Big O then dont get ∞

↳ if Ω then lim > 0.

↳ if Θ then $f(n) = O\,g(n)$ & $g(n) = O\,f(n)$

* $f = o(g) \Rightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} = 0.$

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

* $f = o(g) \rightarrow$ f grows slower than g.

$|f(n)| \leq c \cdot g(n)$

* $f = \omega(g) \rightarrow$ f grows faster than g.

$c \cdot g(n) \leq |f(n)|$

* $f \sim g$: f/g approaches 1

$$\lim_{n \to \infty} \frac{f(n)}{g(n)} = 1$$

Imp

* $f = \Theta(g) \Rightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} \neq 0, \infty$

* $f = O(g) \Rightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} \neq \infty$.

* $f = \Omega(g) \Rightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} \neq 0$.

* $f = \omega(g) \Rightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} = \infty$

Dividing Functions

$) = aT(n/b) + f(n)$

$f(n) = \Theta(n^k \log^p n)$

ase 1: if $\log_b^a > k$ then $\Theta(n^{\log_b^a})$

ase 2: if $\log_b^a = k$

if $p > -1$ $\Theta(n^k \log^{p+1} n)$

if $p = -1$ $\Theta(n^k \log\log n)$

if $p < -1$ $\Theta(n^k)$

Case 3: if $\log_b^a < k$ if $p \geq 0$ $\Theta(n^k \log^p n)$

if $p < 0$ $O(n^k)$