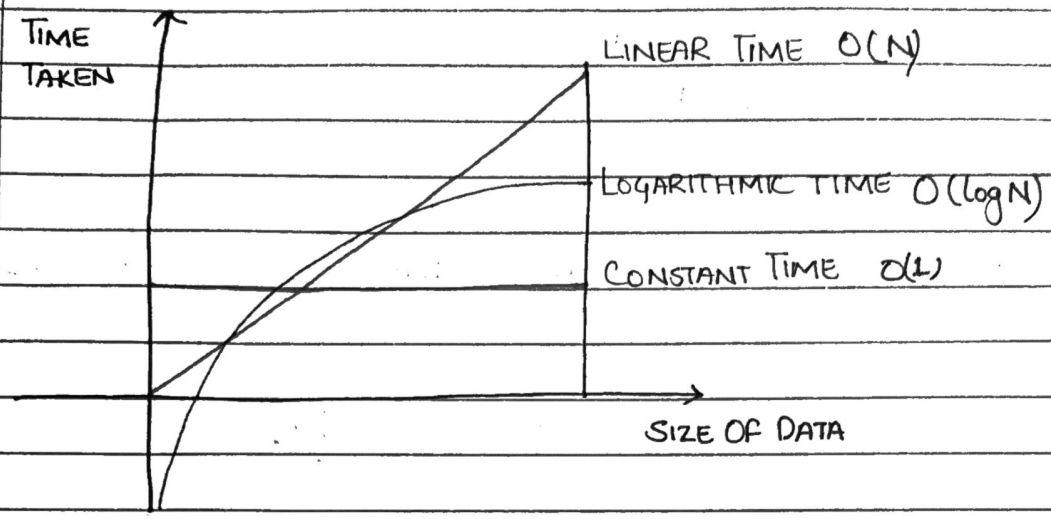
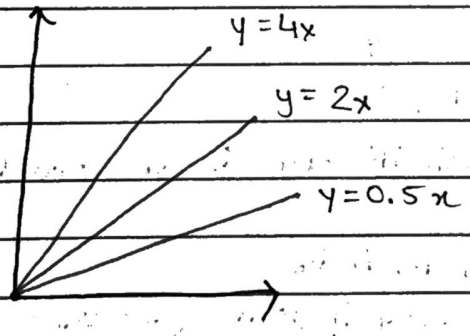
SPACE AND TIME COMPLEXITY



~~O(1)~~ → $O(1) < O(\log N) < O(N)$

- ~~Chec~~ Keep WORST CASES in mind
- Always compare values of time for large amount of data.



- The trend of the slope is taken into consideration (LINEAR)
  Constants are ignored.

Page No.
Date :

* $O(N^3 + \log N)$

↳ Supposedly, Time taken: $(1 \text{ million sec})^3 + \log(1 \text{ million})$

$= (1{,}000{,}000)^3 + 6$

↳ LESS DOMINATING

↳ MORE DOMINATING

∴ IGNORE LESS DOMINATING TERMS WHILE CALCULATING COMPLEXITIES.

BIG-OH NOTATION

- Gives the upper bound for the algorithm, the time taken by it keeping in consideration the worst case scenario.

$O(N^4)$

upper bound

BIG OMEGA NOTATION

- Opposite of BIG OH

↳ Gives the lower-bound for the time/[redacted] taken by the given algorithm.

$\Omega(N^4)$ lower-bound

BIG THETA NOTATION

- The exact behaviour (time/[redacted]) is calculated for specific algorithm by bounding it from the top as well as bottom.

$\Theta(N^2)$

Page No.
Date :

LITTLE - OH NOTATION

- Gives the upper bound of space/time occupied by the algorithm which isn't strictly true.

LITTLE - OMEGA NOTATION

- Gives the lower bound of space/time occupied by the algorithm which isn't strictly true.

SPACE COMPLEXITY

Auxiliary Space - Extra temporary space used by the algorithm

Space complexity of an algorithm is the total space occupied with respect to the input. It is the total space including Auxiliary space and the space used by input.