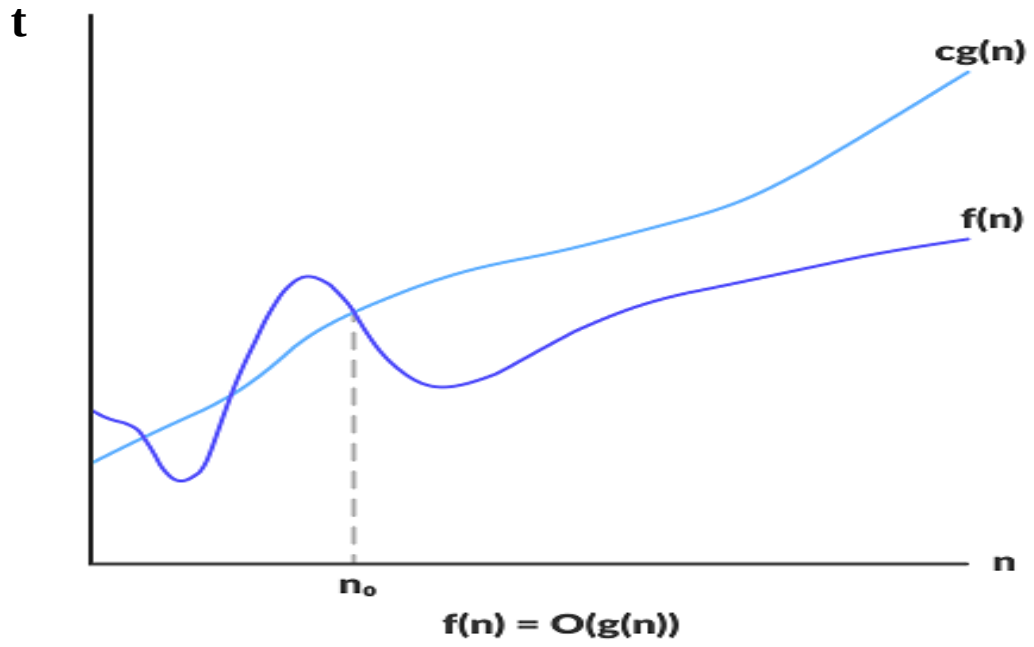
Asymptotic Notations...

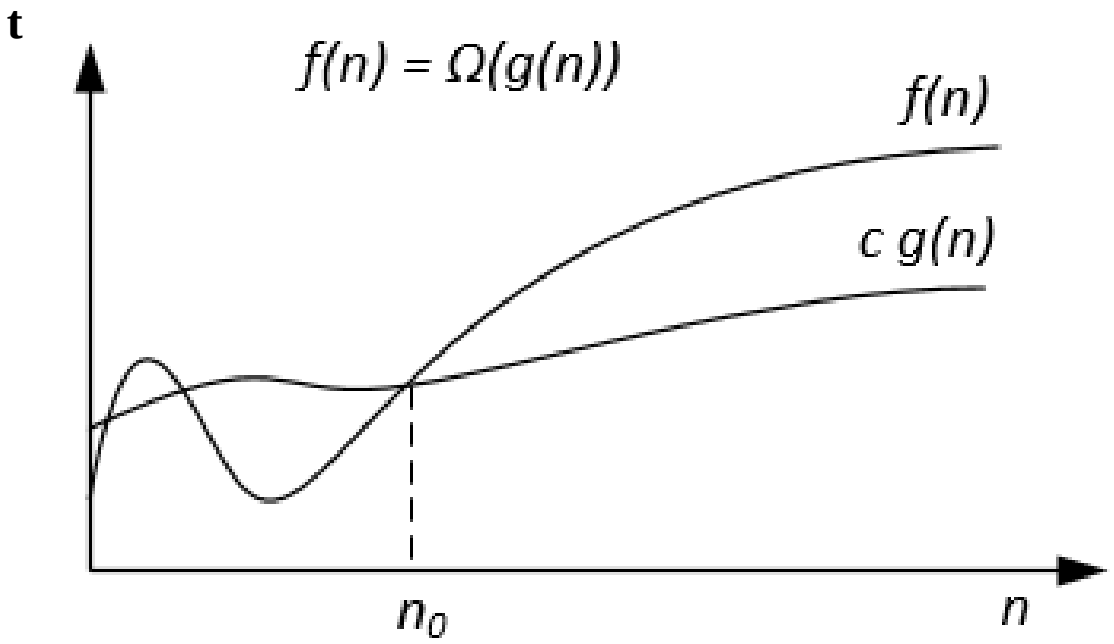
- **Big-Oh Notation (O)**
  - It describes the worst case scenerio,it represents the upper bound running time complexity of an algorithm .



  - if $f(n) \leq cg(n)\ \forall\ n \geq n_0$ where $c>0$ and $n_0 >=1$ then we can say that $f(n) = O(g(n))$.

Asymptotic Notations...

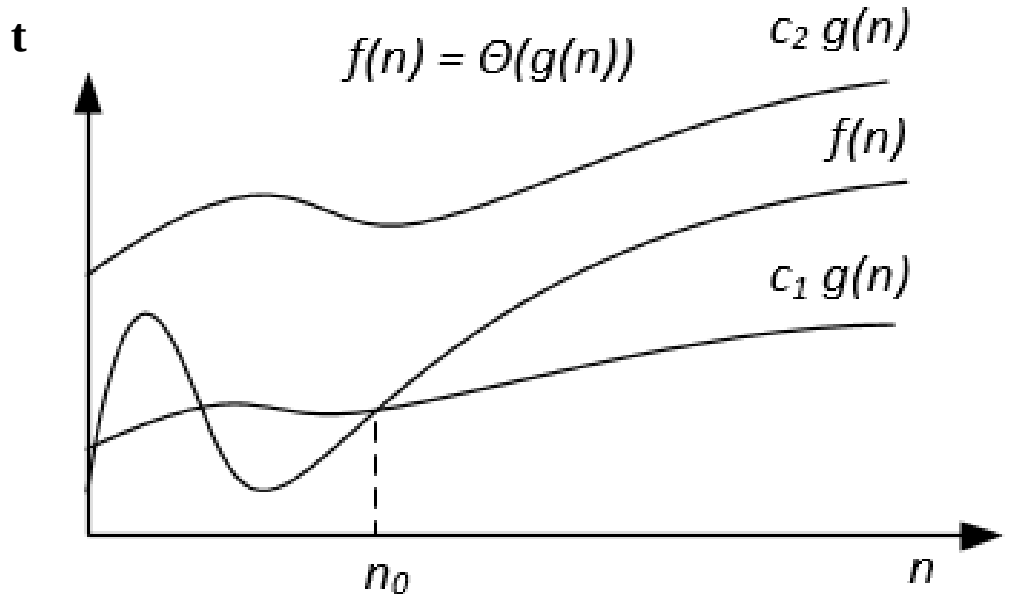
- **Big-Omega Notation(Ω)**
  - It describes the best case scenerio,it represents the lower bound running time complexity of an algorithm .



  - if f(n)>=c.g(n) ∀ n≥n_0 where c>0 and n_0 >=1 then we can say that f(n) = **Ω** (g(n)) .

Asymptotic Notations...

- **Big -Theta Notation (Θ)**
  - It describes the average case scenerio,it represents the lower bound and upper bound of an algorithm .



  - if c1g(n)<=f(n)<=c2.g(n) ∀ n≥n_0 where c1,c2>0 and n_0 >=1 then we can say that f(n) = **Θ** (g(n)) .

Linear & Non-linear Data Structure

➢Linear Data Structure:

- ➢ In this data elements are arranged sequentially or in linear fashion.
- ➢ It involves single level ,therefore we can traverse all the elements in single run only.
- ➢ Linear data structures are easy to implement because computer memory is arranged in a linear way.

➢Non -Linear Data Structure:

- ➢In this data elements are not arranged sequentially or linearly .
- ➢It does not involve single level, therefore, we can't traverse all the elements in single run.
- ➢ They are not easy to implement in comparison to linear data structure.
- ➢ It utilizes computer memory efficiently in comparison to a linear data structure.