1. If limit f(n)/g(n) <= c , then f(n) = O(g(n)) where c is constant greater than 0.

N 🡪 ∞

1. If limit f(n)/g(n) >= c , then f(n) = Ω(g(n)) where c is constant greater than 0.

N 🡪 ∞

1. If limit f(n)/g(n) = c , then f(n) = theta(g(n)) where c is constant greater than 0.

N 🡪 ∞

1. If limit f(n)/g(n) = 0 , then f(n) = small o(g(n)) where c is constant greater than 0.

N 🡪 ∞

1. If limit f(n)/g(n) = ∞ , then f(n) = small omega(g(n)) where c is constant greater than 0.

N 🡪 ∞

**Log method:**

1. If log f(n) = small oh (o) (log g(n)) 🡪 f(n) = o(g(n))
2. If log f(n) = small omega(ῳ) (log g(n)) 🡪 f(n) = ῳ(g(n))

**Be careful:**

1. log f(n) = theta(log g(n)) 🡪 we can’t decide

**About logarithm:**

1. 2x = 16 🡪 log2(16) = x
2. 2log2x = x asymptotically
3. Log n! = nlogn asymptotically

**GATE PROBLEMS:**

1. f(n) = 3n ^

g(n) = 2^2n)

h(n) = n!

Which of the following is true?

1. h (n) = O(f(n)) 🡪 h(n) <= f(n) 🡪 wrong
2. h (n) = O(g(n))🡪 h(n) <= g(n) 🡪 wrong
3. g(n) != O(f(n)) 🡪 g(n) !<= f(n) 🡺 wrong
4. f(n) = O(g(n)) 🡪 f(n) <= g(n) 🡪 right

Solution,

1. g (n) = 2^ ( \* log2n)

= 2(log2n)^🡪 n^

1. f(n) = 3n ^ . So f(n) and g(n) are asymptotically same i.e f(n) = g(n)
2. h(n) = n!

We know,

Log n! = nlogn

🡪 Log(n^ = logn (we have n^ on both f(n) and g(n))

Now, comparing between nlogn and logn

nlogn > logn

So, h(n) > g(n) and g(n) = f(n).s