Asymptotic Notation

Here we all see we need to find the Time Complexity we The Notation like O(n) or o(1) or o(log n) etc…

This is Known as Big O Notation

What Is Asymptotic Notation ?

Running Time Of the Algorithms

Why we need This ?

To Notify the best solution compare to others

Types of Asymptotic Notation :

1. Big O Notation (O)
2. Omega Notation (Ω)
3. Theta Notation (Θ)

Which One Is Preferable ?

Compare to three Big O Notation is Important

Lets see the example 1 :

There is a function called f(x) whose value is 5n

F(x) = 5x

And there is a g(x) whose value is n

G(x) = x

Meaning is there is a dependencies of x

If we say the value of f(x) is equal to order of g(x)

F(x) = o(g(x))

What is this mean ?

It means f(x) is less than c times g(x)

F(x> <= c.g(x)

C 🡪 Constant

Constant > 1 (greater than 0 ) in Big 0

What is this mean ?

If we put any value which should be greater than or equal to g(x)

Where we use c = 5

5x <= 5\*g(x)

Now f(x) <= cg(x)

Proved

We see Another example 2:

F(x) = x

G(x) = 5x

Now

F(x) <= c\*g(x)

Becozz if we give c value has ½ c=1/2

Now c\*g(x) is greater than f(x)

Proved

Example 3 :

F(x) = x

G(x) = x2 (x square)

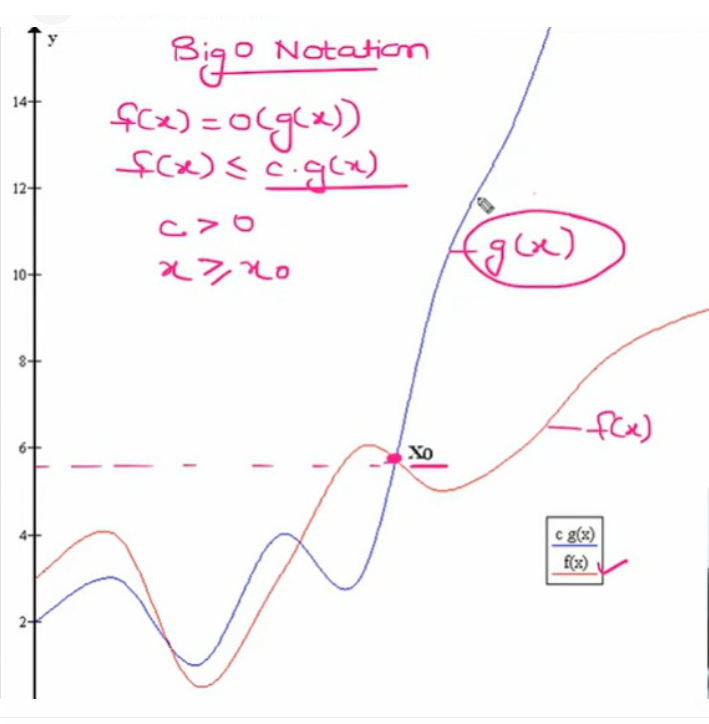
What is the value to prove our condition

We use c = 1/x

Our Condition is False Not Proved

Becozz we give 1/x now our c is not constant it is dependent on our x value , So it is False

Now we See the Bio O Notation In Graph :



Here we see f(x) increase and decrease at some point (5.8) g(x) keep increasing constantly

So Big O Notation we say it has Worst Case Scenario

Worst Case Scenario means if our date is huge it works good

Ex :

For I in range(0 , n)

Now our n is how huge number it works good in Big O Notation

That’s why we Use Big 0 Notation O( ) In Everywhere