Asymtotic Notations

Mohammed Rizin
Unemployed

March 21, 2025

1 Types of Asymptotic Notations

In last Section we learn the Types of Time function:

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

The type of Asymptotic Notations are:

1. $O(n)$	▶ Big Oh	▶ Upper Bound
2. $\Omega(n)$	▶ Big Omega	▶ Lower Bound
3. $\Theta(n)$	▶ Theta	▶ Average Time Function (useful)

1.1 Big Oh Notation

The function $f(n) = O(g(n)) \leftrightarrow \quad \exists$ +ve constant c and n_0 such that $f(n) \leq c \cdot g(n) \quad \forall n \geq n_0$

For eg:

$$\begin{aligned} f(n) &= 2n + 3 \\ 2n + 3 &\leq 10n \;\; \forall n \geq 1 \\ f(n) &\leq 10 \cdot g(n) \\ \text{The Time Complexity : } f(n) &= O(n) \end{aligned}$$

$$\begin{aligned} f(n) &= 2n + 3 \\ 2n + 3 &\leq 2n + 3n \\ 2n + 3 &\leq 5n \;\; \forall n \geq 1 \\ f(n) &\leq 5 \cdot g(n) \\ \text{Still The Time Complexity : } f(n) &= O(n) \end{aligned}$$

$$\begin{aligned} f(n) &= 2n + 3 \\ \text{So Can I write:} & \\ 2n + 3 &\leq 2n^2 + 3n^2 \\ 2n + 3 &\leq 5 \cdot n^2 \;\; \forall n \geq 1 \\ f(n) &\leq 5 \cdot g(n) \\ \text{Yes, Then the Time Complexity : } f(n) &= O(n^2) \end{aligned}$$

So I can say that $f(n) = O(n)$, $f(n) = O(n^2)$, $f(n) = O(n^3)$, $f(n) = O(2^n)$, $f(n) = O(n!)$, etc. But we cannot say $f(n) = O(\log n)$, $f(n) = O(\sqrt{n})$, $f(n) = O(1)$, etc.

¿ Now a question arises, Why do I need to write all those if we know that $f(n) = O(n)$? Yeah, When we write Big Oh, we write the closest upper bound. When you any above that, it is also correct. But not usefulYeah, When we write Big Oh, we write the closest upper bound. When you any above that, it is also correct. But not useful.

1.2 Big Omega Notation

The function $f(n) = \Omega(g(n)) \leftrightarrow \quad \exists$ +ve constant c and n_0 such that $f(n) \geq c \cdot g(n) \quad \forall n \geq n_0$

For eg:

$$f(n) = 2n + 3$$
$$2n + 3 \geq 1n \ \ \forall n \geq 1$$
$$f(n) \geq \underbrace{1}_{c} \cdot \underbrace{n}_{g(n)}$$
$$\text{The Time Complexity : } f(n) = \Omega(n)$$

$$f(n) = 2n + 3$$
$$2n + 3 \geq 2n + 3$$
$$2n + 3 \geq 5n \ \ \forall n \geq 1$$
$$f(n) \geq 5 \cdot g(n)$$
$$\text{Still The Time Complexity : } f(n) = \Omega(\log n)$$

$$f(n) = 2n + 3$$
$$\text{So Can I write:}$$
$$2n + 3 \geq 2n^2 + 3n^2$$
$$2n + 3 \geq 5 \cdot n^2 \ \ \forall n \geq 1$$
$$f(n) \geq 5 \cdot g(n)$$
$$\text{No, the Time Complexity can't be: } f(n) = \Omega(n^2)$$

So which one is correct? $f(n) = \Omega(n)$ or $f(n) = \Omega(\log n)$? Both are correct. But the closest lower bound is

$$\boxed{f(n) = \Omega(n)}$$

1.3 Theta Notation

The function $f(n) = \Theta(g(n)) \leftrightarrow \quad \exists$ +ve constant c_1, c_2 and n_0 such that $c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n) \quad \forall n \geq n_0$

For eg:

$$f(n) = 2n + 3$$
$$1n \leq 2n + 3 \leq 5n \ \ \forall n \geq 1$$
$$\underbrace{1}_{c1} \cdot \underbrace{n}_{g(n)} \leq f(n) \leq \underbrace{5}_{c2} \cdot \underbrace{n}_{g(n)}$$

$g(n)$ on both sides is n.So, the Time Complexity is: $f(n) = \Theta(n)$

$$f(n) = 2n + 3$$
$$\underbrace{1n^2}_{\text{X}} \leq 2n + 3 \leq 5n^2 \ \ \forall n \geq 1$$
$$\text{So, the only Time Complexity : } \boxed{f(n) = \Theta(n)}$$

Since I cannot find the any other $g(n)$ which satisfies the above equation, I can say that the Time Complexity is $\boxed{\Theta(n)}$.

1.4 Some Myths

People say that $f(n) = O(n)$ represent the "Worst" case. But it is not true. It is the "Upper Bound". And $f(n) = \Omega(n)$ represent the "Best" case. But it is not true. It is the "Lower Bound".
Don't get confused with the "Best" and "Worst" case with "Lower" and "Upper" bound.

1.5 Example 1

$$\begin{aligned} f(n) &= n! \\ f(n) &= 1 \cdot 2 \cdot 3 \cdot 4 \cdots n \\ 1 \cdot 1 \cdot 1 \cdot 1 \cdots 1 \leq 1 \cdot 2 \cdot 3 &\cdots n \leq n \cdot n \cdot n \cdots n \ \ \forall n \geq 1 \\ 1 \leq n! &\leq n^n \end{aligned}$$

So, we have the Upper Bound Time Complexity : $\boxed{f(n) = O(n^n)}$
And we have the Lower Bound Time Complexity : $\boxed{f(n) = \Omega(1)}$
There is no $g(n)$ which satisfies the above equation. So, the Average Time Complexity doesn't exist.
The best Asymptotic Notation is $\Theta(n)$. Because it gives the average time function. It is useful. Suppose you are buying a mobile phone and the shopkeeper says that there are phones between 2000 and 200000 even though you said that you want a phone for 20000. 2000 is the lower bound and 200000 is the upper bound. Are they wrong ? No. But it is not useful. So, the best Asymptotic Notation is $\Theta(n)$. If you are telling me the Big Oh , i don't know whether it is nearest upper bound or not. So, it is not useful. So, the best Asymptotic Notation is $\Theta(n)$. Big Oh and Big Omega are mainly useful in case of $n!$.

1. The function $f(n)$ belong to the n class. So everything above it is upper bound. So the upper bound of $f(n)$ could be $O(n)$, $O(n^2)$, $O(n^3)$, $O(2^n)$, $O(n!)$, etc.

2. The function $f(n)$ belong to the n class. So everything below it is lower bound. So the lower bound of $f(n)$ could be $\Omega(n)$, $\Omega(\sqrt{n})$, $\Omega(\log n)$, $\Omega(1)$, etc.

3. The function $f(n)$ belong to the n class. So everything equal to it is average bound. So the average bound of $f(n)$ could be $\Theta(n)$.

2 Properties of Asymptotic Notations

2.1 General Properties

1. if $f(n)$ is $O(g(n))$ then $a * f(n)$ is $O(g(n))$ for any constant $a > 0$.
   eg: $f(n) = 2n^2 + 3$ is $O(n^2)$
   then $2f(n) = 4n^2 + 6$ is $O(n^2)$
   This is true for all three Asymptotic Notations.

2. Reflexive Property: if $f(n)$ is given then $f(n)$ is $O(f(n))$
   eg : $f(n) = n^2$ is $O(n^2)$
   This is true for all three Asymptotic Notations.

3. Transitive Property: if $f(n)$ is $O(g(n))$ and $g(n)$ is $O(h(n))$ then $f(n)$ is $O(h(n))$
   eg: $f(n) = n^2$ is $O(n^3)$ and n^3 is $O(n^4)$ then $f(n)$ is $O(n^4)$
   This is true for all three Asymptotic Notations.

4. Symmetric Property: if $f(n)$ is $\Theta(g(n))$ then $g(n)$ is $\Theta(f(n))$
   eg: $f(n) = n^2$ is $\Theta(n^2)$ then n^2 is $\Theta(n^2)$
   This is true for only Θ Asymptotic Notation.

5. Transpose Symmetric: if $f(n)$ is $O(g(n))$ then $g(n)$ is $\Omega(f(n))$
   eg: $f(n) = n^2$ is $O(n^3)$ then n^3 is $\Omega(n^2)$
   This is true for both O and Ω Notations.

6. if $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$ then

$$\overbrace{g(n)}^{\downarrow} \leq f(n) \leq \overbrace{g(n)}^{\downarrow}$$
$$\boxed{f(n) = \Theta(g(n))}$$

7. if $f(n) = O(g(n))$ and $d(n) = O(e(n))$ then $f(n) \pm d(n) = O(max(g(n), e(n)))$
   eg:
   $f(n) = n^2$ is $O(n^2)$
   and $d(n) = n^3$ is $O(n^3)$
   then $f(n) \pm d(n) = n^2 + n^3 is O(n^3)$

8. if $f(n) = O(g(n))$ and $d(n) = O(e(n))$ then $f(n) \cdot d(n) = O(g(n) \cdot e(n))$
   eg:
   $f(n) = n^2$ is $O(n^2)$
   and $d(n) = n^3$ is $O(n^3)$
   then $f(n) \cdot d(n) = n^2 \cdot n^3 is O(n^5)$

3 Comparison of functions

For eg : Between n^2 and n^3 which one is greater? 1. Method 1: Sample Method

$$\begin{aligned} n = 1 \rightarrow 1^2 = 1 \text{ and } 1^3 = 1 \\ n = 2 \rightarrow 2^2 = 4 \text{ and } 2^3 = 8 \\ n = 3 \rightarrow 3^2 = 9 \text{ and } 3^3 = 27 \\ n = 4 \rightarrow 4^2 = 16 \text{ and } 4^3 = 64 \\ n = 5 \rightarrow 5^2 = 25 \text{ and } 5^3 = 125 \end{aligned}$$

So, n^3 is greater than n^2.
2. Method 2: Apply Log on both sides

$$\begin{gathered} n^2 \leq n^3 \\ \log n^2 \leq \log n^3 \\ 2 \log n \leq 3 \log n \\ 2 \leq 3 \end{gathered}$$

So, n^3 is greater than n^2.
Logarithm Properties:

$$\begin{aligned} \log ab &= \log a + \log b \\ \log \frac{a}{b} &= \log a - \log b \\ \log a^b &= b \log a \\ a^{\log_c b} &= b^{\log_c a} \\ a^b = n &\rightarrow b = \log_a n \end{aligned}$$

For Example:
Which is greater $n^2 \log n$ or $n(\log n)^{10}$?

$$\begin{gathered} n^2 \log n \ \mid \ n(\log n)^{10} \\ 2 \log n + \log(\log n) \ \mid \ \log n + 10 \log(\log n) \\ \text{Bigger term is } \log n \text{ and } \log(\log n) \text{ becomes negligible} \\ 2 \log n \geq \log n \end{gathered}$$

So, $n^2 \log n$ is greater than $n(\log n)^{10}$.

$$\text{Which is greater } f(n) = 3n^{\sqrt{n}} \text{ or } 2^{\sqrt{n} \cdot \log_2 n}?$$

$$\begin{aligned} 3n^{\sqrt{n}} &\mid 2^{\sqrt{n} \cdot \log_2 n} \\ 3n^{\sqrt{n}} &\mid 2^{\log_2 n^{\sqrt{n}}} \\ 3n^{\sqrt{n}} &\mid n^{\sqrt{n}^{\log_2 2 = 1}} \\ 3n^{\sqrt{n}} &\simeq n^{\sqrt{n}} \end{aligned}$$

Asymtotically both are equal.

Which is greater $f(n) = 2^n$ or 2^{2n}?

$$2^n \mid 2^{2n}$$

Applying log on both sides,

$$n < 2n$$

Asymtotically we cannot say both are equal after applying log

Another Example:

$$f_1(n) = \begin{cases} n^3 & n < 100 \\ n^2 & n \geq 100 \end{cases}$$

$$f_2(n) = \begin{cases} n^2 & n < 10000 \\ n^2 & n \geq 10000 \end{cases}$$

Which is greater $f_1(n)$ or $f_2(n)$?

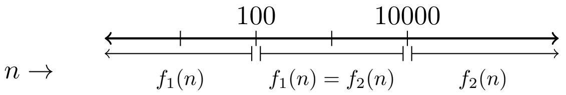

Figure 1: Comparison of $f_1(n)$ and $f_2(n)$

Since $f_2(n)$ is greater than $f_1(n)$ $\forall n \geq 10000$. So, $f_2(n)$ is greater than $f_1(n)$.

4 Exercises

True or False:

1. $(n + k)^m = \Theta(n^m)$
2. $2^{n+1} = O(2^n)$
3. $2^{2n} = O(2^n)$
4. $\sqrt{\log n} = O(\log(\log n))$
5. $n^{\log n} = O(2^n)$

Answers:

1. True
   According to Binomial Expansion, the highest degree term is n^m.
   So, $(n + k)^m = \Theta(n^m)$
2. True
   $2^{n+1} = 2 \cdot 2^n \simeq O(2^n)$ which is also upper bound.
3. False
   $2^{2n} = 4^n \simeq O(4^n)$
   2^n is not upper bound of 4^n.
4. False
   $\sqrt{\log n} = \log(\log n)$
   $\frac{1}{2}\log n = \log\log\log n$
   $\log n \geq \log\log\log n$ This is not upper bound.
5. True
   $n^{logn} = 2^n$
   $\log^2 n = n$
   $\log n = \sqrt{n}$
   $\log logn = \frac{1}{2}\log n$
   So $\log\log n \leq \log n$
   i.e. $n^{logn} = O(2^n)$