Text:

1. Operations and properties of asymptotic notation

We have already seen how asymptotic notation can be used within mathematical formulas. For example, in introducing O-notation, we wrote "$n = O(n^2)$." We might also write $2n^2 + 3n + 1 = 2n^2 + \Theta(n)$. How do we interpret such formulas?

When the asymptotic notation stands alone on the right-hand side of an equation, as in $n = O(n^2)$, we have already defined the equal sign to mean set membership: $n \in O(n^2)$. In general, however, when asymptotic notation appears in a formula, we interpret it as standing for some anonymous function that we do not care to name. For example, the formula $2n^2 + 3n + 1 = 2n^2 + \Theta(n)$ means that $2n^2 + 3n + 1 = 2n^2 + f(n)$, where $f(n)$ is some function in the set $\Theta(n)$. In this case, $f(n) = 3n + 1$, which indeed is in $\Theta(n)$.

Using asymptotic notation in this manner can help eliminate inessential detail and clutter in an equation. For example, the worst-case running time of merge sort as the recurrence can be written as $T(n) = 2T(n/2) + \Theta(n)$. If we are interested only in the asymptotic behaviour of $T(n)$, there is no point in specifying all the lower-order terms exactly; they are all understood to be included in the anonymous function denoted by the term $\Theta(n)$.

The number of anonymous functions in an expression is understood to be equal to the number of times the asymptotic notation appears. For example, in the expression $\sum_{i=1}^{n} O(i)$, there is only a single anonymous function (a function of i). This expression is thus *not* the same as $O(1) + O(2) + \ldots + O(n)$, which doesn't really have a clean interpretation.

In some cases, asymptotic notation appears on the left-hand side of an equation, as in $2n^2 + \Theta(n) = \Theta(n^2)$. We interpret such equations using the following rule: *No matter how the anonymous functions are chosen on the left of the equal sign, there is a way to choose the anonymous functions on the right of the equal sign to make the equation valid*. Thus, the meaning of our example is that for any function $f(n) \in \Theta(n)$, there is some function $g(n) \in \Theta(n^2)$ such that $2n^2 + f(n) = g(n)$ for all n. In other words, the right-hand side of an equation provides coarser level of detail than the left-hand side.

A number of such relationships can be chained together, as in

$$\begin{aligned} 2n^2 + 3n + 1 &= 2n^2 + \Theta(n) \\ &= \Theta(n^2). \end{aligned}$$

We can interpret each equation separately by the rule above. The first equation says that there is some function $f(n) \in \Theta(n)$ such that $2n^2 + 3n + 1 = 2n^2 + f(n)$ for all n. The second equation says that for any function $g(n) \in \Theta(n)$ (such as the $f(n)$ just mentioned), there is some function $h(n) \in \Theta(n^2)$ such that $2n^2 + g(n) = h(n)$ for all n. Note that this interpretation implies that $2n^2 + 3n + 1 = \Theta(n^2)$, which is what the chaining of equations intuitively gives us.

1.1. Properties of Asymptotic Notations:

To understand the concept of asymptotic notations fully, let us look at some of the general properties of the asymptotic notations.

1.1.1. General:

If f(n) = O (g(n)), then

a × f(n) = O(g(n))

e.g. $f(n) = 2n^2+5$ *is* $O(n^2)$, then

$4f(n) = 8n^2+20$ *is also* $O(n^2)$

1.1.2. Reflexive:

Given f(n), then ***f(n) = O(f(n))***

e.g. $f(n)= n^2 = O(n^2)$

i.e. every function is an upper-bound of itself.

1.1.3. Transitive:

If f(n) = O (g(n)) and g(n) = O (h(n)), then

f(n) = O (h(n))

e.g. $f(n) = n$, $g(n) = n^2$, & $h(n)=n^3$

here, $f(n)=O(n^2)$ & $g(n) = O(n^3)$

=> $f(n) = O(n^3)$ *or O(h(n))*

1.1.4. Symmetric:

If ***f(n)*** = ***Θ(g(n)), then g(n)*** = ***Θ(f(n))***

e.g. $f(n) = n^2$ & $g(n) = n^2+3$

$f(n)= \Theta(n^2)$ & $g(n)= \Theta(n^2)$

1.1.5. Transpose Symmetric:

i. *If f(n) = O(g(n)), then g(n) = Ω(f(n))*

e.g. $f(n)= n$ & $g(n) = n^2$

Here, $f(n) = O(n^2)$ & $g(n) = \Omega(n)$

ii. *If f(n) = O(g(n)) & d(n) = O(e(n)), then*

f(n) + d(n) = O(max(g(n),e(n)))

e.g. $f(n) = n = O(n)$

$d(n) = n^2 = O(n^2)$

$f(n) + d(n) = n + n^2 = O(n^2)$

iii. *if f(n) = O(g(n)) & d(n) = O(e(n)), then*

f(n) × d(n) = O(g(n) × e(n))

1.2. Best, Average, and Worst Cases:

Often times, we confuse the best, average and worst time complexities with the asymptotic notations. However, these are not always equal to O, Ω, & Θ. Let us try to understand the difference with the help of an example.

- Worst-case: The maximum number of steps taken on any instance of size *a*.
- Best-case: The minimum number of steps taken on any instance of size *a*.

- Average case: An average number of steps taken on any instance of size *a*.

Example:

i. In Linear Search, we have

A. Best Case: Key element at 1st Index

Best Case Time B(1)

B. Worst Case: Key element at nth Index (or not present).

Worst Case Time W(n)

C. Average Case: All possible cases/Number of cases

$= (1 + 2 + 3 + \dots + n) / n$

$= [\, n(n+1)/2 \,] / n = (n+1)/2$

Average Case Time A(n)

Therefore, $B(n) = O(1), \Omega(1), \& \Theta(1)$

$W(n) = O(n), \Omega(n), \& \Theta(n)$

$A(n) = O(n), \Omega(n), \& \Theta(n)$

ii. Binary Search Tree:

A. Case 1: Height Balanced Tree

$B(n) = 1$ (Key element present at the root node)

$W(n) = \log n$ (Key element present at the leaf node or not present at all)

B. Case 2: Skewed Tree

$W_{max}(n) = n$