Complexity Analysis

Big-O Notation

This operation means “order of” and is used to describe the general behaviour of an algorithm.

We can calculate the complexity of an algorithm by many delimeters. One common way is to count the number of in terms your input, and the complexity will be the greatest power of ‘n’ in that equasion.

Synbols of Complexity

O - The “order of” operator, defines worst case complexity.
Θ - Defines the general average complexity.
Ω - Defines best-case complexity.

Properties of Big-O

1. If $f(n) = O(g(n))$ and $g(n)$ is $O(h(n))$ then $f(n) = O(h(n))$.
2. If $f(n) = O(h(n)))$ and $g(n) = O(h(n))$ then $f(n) + g(n)$ is $O(h(n))$.
3. $an^k = O(n^k)$.
4. $n^k = O(n^{k+j})$ for any $j > 0$.
5. If $f(n) = c.g(n)$ then $f(n) = O(g(n))$.
6. $\log_a n = O(\log_b n)$ for any $a,b >1$
7. $\log_a n$ is $O(\log_2 n)$ for any positive $a \neq 1$.

Regan Koopmans, 2016