

Pilas

Árboles

List

Colas

Análisis Asintótico

Stack

Array

Tree

Queue

Listas

Análisis Asintótico

El análisis asintótico de un algoritmo se refiere a la definición de la base/encuadre matemático de su rendimiento en tiempo de ejecución. Usando el análisis asintótico, podemos concluir muy bien el mejor caso, el caso promedio y el peor escenario de un algoritmo.

El análisis asintótico está ligado a la entrada, es decir, si no hay entrada para el algoritmo, se concluye que funciona en un tiempo constante. Aparte de la "entrada", todos los demás factores se consideran constantes.

El análisis asintótico se refiere a calcular el tiempo de ejecución de cualquier operación en unidades matemáticas de cálculo. Por ejemplo, el tiempo de ejecución de una operación se calcula como $f(n)$ y puede ser para otra operación se calcula como $g(n^2)$. Esto significa que el tiempo de ejecución de la primera operación aumentará linealmente con el aumento de n y el tiempo de ejecución de la segunda operación aumentará exponencialmente cuando n aumente. De manera similar, el tiempo de ejecución de ambas operaciones será casi el mismo si n es significativamente pequeño.

Por lo general, el tiempo requerido por un algoritmo se divide en tres tipos:

- Mejor caso: tiempo mínimo requerido para la ejecución del programa.
- Caso promedio: tiempo promedio requerido para la ejecución del programa.
- Peor de caso: tiempo máximo requerido para la ejecución del programa.

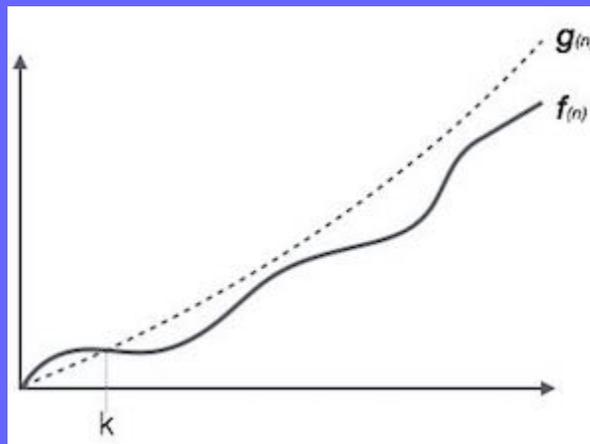
A continuación se muestran las notaciones asintóticas de uso común para calcular la complejidad del tiempo de ejecución de un algoritmo.

- Notación O
- Notación Ω
- Notación θ



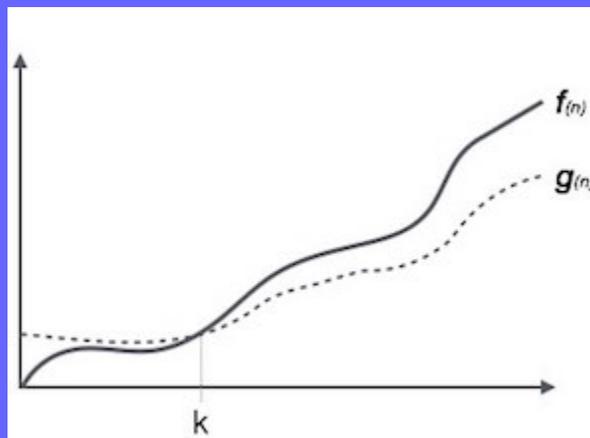
Notación de Big Oh, O

La notación $O(n)$ es la forma formal de expresar el límite superior del tiempo de ejecución de un algoritmo. Mide la complejidad del tiempo en el peor de los casos o la mayor cantidad de tiempo que un algoritmo puede tardar en completarse.



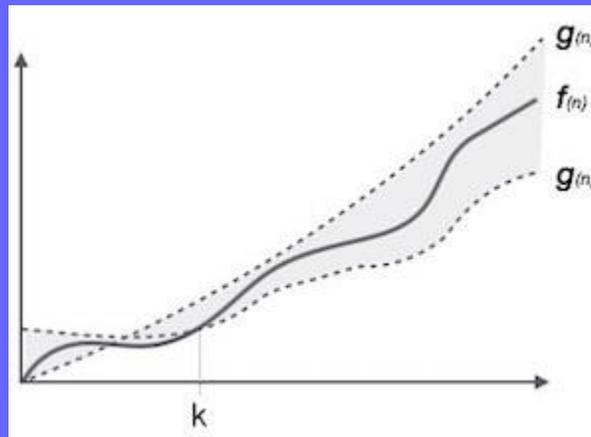
Notación Omega, Ω

La notación $\Omega(n)$ es la forma formal de expresar el límite inferior del tiempo de ejecución de un algoritmo. Mide la mejor complejidad del tiempo del caso o la mejor cantidad de tiempo que un algoritmo puede tardar en completarse.



Notación theta, θ

La notación $\theta(n)$ es la forma formal de expresar tanto el límite inferior como el límite superior del tiempo de ejecución de un algoritmo. Se representa de la siguiente manera:



Notaciones asintóticas comunes

A continuación se muestra una lista de la complejidad del tiempo del algoritmo que ocurre comúnmente en orden creciente:

Name	Notation
Constant	$O(1)$
Logarithmic	$O(\log n)$
Linear	$O(n)$
N-LogN	$O(n \log n)$
Quadratic	$O(n^2)$
Polynomial	$O(n^c)$ c is a constant & $c > 1$
Exponential	$O(c^n)$ c is a constant & $c > 1$
Factorial or N-power-N	$O(n!)$ or $O(n^n)$