

Growth of Functions

Alexandre David

Asymptotic Notations

- Asymptotic running time of algorithms

$n \rightarrow \infty$

- θ -notation: *asymptotic tight bound*

$$\theta(g(n)) = \{f(n) : \exists c_1 > 0 \ c_2 > 0 \ n_0 \geq 0, \forall n \geq n_0, 0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)\}$$

$\theta(g(n))$ is a set, so we write $f(n) \in \theta(g(n))$

- O -notation: *asymptotic upper bound*

$$O(g(n)) = \{f(n) : \exists c > 0 \ n_0 \geq 0, \forall n \geq n_0, 0 \leq f(n) \leq cg(n)\}$$

weaker than θ : $O(g(n)) \subseteq \theta(g(n))$

- Ω -notation: *asymptotic lower bound*

$$\Omega(g(n)) = \{f(n) : \exists c > 0 \ n_0 \geq 0, \forall n \geq n_0, 0 \leq cg(n) \leq f(n)\}$$

weaker than θ : $\Omega(g(n)) \subseteq \theta(g(n))$

- Theorem: $f(n) = \theta(g(n))$ iff $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$

Asymptotic Notations

- o-notation: upper bound not asymptotically tight.

$o(g(n)) = \{f(n) : \forall c > 0, \exists n_0 > 0. \forall n \geq n_0, 0 \leq f(n) < cg(n)\}$

$2n = o(n^2)$ but $2n^2 \neq o(n^2)$

- ω-notation: lower bound not asymptotically tight.

$\omega(g(n)) = \{f(n) : \forall c > 0, \exists n_0 > 0. \forall n \geq n_0, cg(n) < f(n)\}$

$n^2/2 = \omega(n)$ but $n^2/2 \neq \omega(n^2)$

- Properties p49

Standard Notations and Common Functions

- Check section 3.2. You should know it, good remainder. Ask questions.
- Fibonacci numbers

