csce350 — Data Structures and Algorithms
Fall 2019 — Lecture Notes: Asymptotic Notations

This document contains slides from the lecture, formatted to be suitable for printing or individual reading, and with occasional supplemental explanations added. It is intended as a supplement to, rather than a replacement for, the lectures themselves — you should not expect the notes to be self-contained or complete on their own.

1 Big-Θ Notation

Basic idea of Big-Θ notation: A way of saying that two functions have the same order of growth.

2 Definition of Θ

Given two functions $t(n)$ and $g(n)$, we say that $t(n) \in \Theta(g(n))$ if there exist positive constants c_1, c_2, and n_0 such that

$$c_2 g(n) \leq t(n) \leq c_1 g(n)$$

for all $n \geq n_0$.

3 Example

Prove, using the definition, that $5n^2 + n + 20 \in \Theta(n^2)$.

Solution:
Choose c_1 =______, c_2 =______, n_0 =______.

$$\begin{aligned} 5n^2 + n + 20 &\leq 5n^2 + n + n && \text{[when } n \geq 20\text{]} \\ &= 5n^2 + 2n \\ &\leq 5n^2 + n \cdot n && \text{[when } n \geq 2\text{]} \\ &= 6n^2 \end{aligned}$$

$$\begin{aligned} 5n^2 + n + 20 &\geq 5n^2 + n \\ &\geq 5n^2 && \text{[when } n \geq 0 \end{aligned}$$

Therefore, $5n^2 \leq 5n^2 + n + 20 \leq 6n^2$ for all $n \geq 20$, so $5n^2 + n + 20 \in \Theta(n^2)$.

4 Example

Prove, using the definition, that $\log(n^2 + n) \in \Theta(\log n)$.

Solution:
Choose c_1 =______, c_2 =______, n_0 =______.

$$\begin{aligned}
\log(n^2+n) &\le \log(n^2+n^2) \qquad [\text{when } n \ge 1] \\
&= \log(2n^2) \\
&= \log 2 + 2\log n \\
&\le \log n + 2\log n \qquad [\text{when } n \ge 2] \\
&= 3\log n
\end{aligned}$$

$$\begin{aligned}
\log(n^2+n) &\ge \log(n^2) \qquad [\text{when } n \ge 0] \\
&= 2\log n
\end{aligned}$$

Therefore, $2\log n \le \log(n^2+n) \le 3\log n$ for all $n \ge 2$, so $\log(n^2+n) \in \Theta(\log n)$.

5 Weaker alternative to Θ: Big-O

Basic idea of Big-O notation: A way of saying that one function has the **same or smaller** order of growth.

Given two functions $t(n)$ and $g(n)$, we say that $t(n) \in O(g(n))$ if there exist positive constants c and n_0 such that

$$t(n) \le cg(n)$$

for all $n \ge n_0$.

6 Weaker alternative to Θ: Big-Ω

Basic idea of Big-Ω notation: A way of saying that one function has the **same or larger** order of growth.

Given two functions $t(n)$ and $g(n)$, we say that $t(n) \in \Omega(g(n))$ if there exist positive constants c and n_0 such that

$$cg(n) \le t(n)$$

for all $n \ge n_0$.

7 Shortcut #1: Limits

Let $L = \lim_{n\to\infty} \frac{t(n)}{g(n)}$.

- If $L = 0$, then $t(n) \in O(g(n))$.
- If $0 < L < \infty$, then $t(n) \in \Theta(g(n))$.
- If $L = \infty$, then $t(n) \in \Omega(g(n))$.

8 Example with limits

Compare the growth rates of $\log_2 n$ and $\sqrt{n}$.

Solution:

$$\begin{aligned}\lim_{n\to\infty}\frac{\log_2 n}{\sqrt{n}} &= \lim_{n\to\infty}\frac{(\log_2 e)(\ln n)}{n^{1/2}}\\ &= \lim_{n\to\infty}\frac{(\log_2 e)(n^{-1})}{(1/2)\cdot n^{-1/2}}\\ &= 2\cdot\log_2 e\cdot\lim_{n\to\infty} n^{-1/2}\\ &= 0\end{aligned}$$

This implies, via Shortcut #1, that $\log_2 n \in O(\sqrt{n})$.

9 Shortcut #2: Polynomials

All polynomials of the same degree have the same order of growth.

$$a_k n^k + a_{k-1} n^{k-1} + \cdots + a_1 n + a_0 \in \Theta(n^k)$$

10 Shortcut #3: Logarithms

Logarithms of all bases greater than 1 have the same order of growth.

For any $a > 1$ and $b > 1$,

$$\log_a n \in \Theta(\log_b n)$$

11 Shortcut #4: Addition

If $t_1(n) \in \Theta(g_1(n))$ and $t_2(n) \in \Theta(g_2(n))$, then

$$t_1(n) + t_2(n) \in \Theta(\max(g_1(n), g_2(n))).$$

(In a sum, just keep the largest order of growth.)

12 Non-exhaustive list of common efficiency classes

- $\Theta(1)$ — constant
- $\Theta(\log n)$ — logarithmic
- $\Theta(n)$ — linear
- $\Theta(n \log n)$ — "en-log-en"

- $\Theta(n^2)$ — quadratic
- $\Theta(n^3)$ — cubic
- $\Theta(2^n)$ — exponential
- $\Theta(n!)$ — factorial