CS 5720 Design and Analysis of Algorithms
Homework #1

Raja Kantheti

September 12, 2024

Problem 1

Determine the order of growth $\Theta(\cdot)$ for each of the following functions:

(a) $(n^2 + 1)^{10}$

Dominant term: $(n^2)^{10} = n^{20}$
Order of growth: $\Theta(n^{20})$

(b) $\sqrt{10n^2 + 7n + 3}$

Dominant term: $\sqrt{10n^2} = n\sqrt{10} \approx n$
Order of growth: $\Theta(n)$

(c) $2n \log((n + 2)^2) + (n + 2)^2 \log(n/2)$

First term analysis:

$$\begin{aligned} 2n \log((n + 2)^2) &= 2n \cdot 2 \log(n + 2) \\ &= 4n \log(n + 2) \end{aligned}$$

Dominant term: $4n \log n$

Second term analysis:

$$\begin{aligned} (n + 2)^2 \log(n/2) &= (n^2 + 4n + 4) \log(n/2) \\ &\approx n^2 \log n \end{aligned}$$

Dominant term: $n^2 \log n$

Overall dominant term: $n^2 \log n$
Order of growth: $\Theta(n^2 \log n)$

(d) $2^{n+1} + 3^{n-1}$

Dominant term: $2^{n+1} \approx 2^n$ and $3^{n-1} \approx 3^n$
Since 3^n grows faster than 2^n,
Order of growth: $\Theta(3^n)$

(e) $\lfloor \log_2 n \rfloor$

Order of growth: $\Theta(\log n)$

Problem 2

Prove the following assertions by using the definitions of the notations involved, or disprove them by giving a specific counterexample.

(a) If $t(n) \in O(g(n))$, then $g(n) \in \Omega(t(n))$.

Proof:
By definition of Big-O, $t(n) \in O(g(n))$ means there exist constants $c > 0$ and n_0 such that $t(n) \leq c \cdot g(n)$ for all $n \geq n_0$.
By definition of Big-Omega, $g(n) \in \Omega(t(n))$ means there exist constants $c' > 0$ and n'_0 such that $g(n) \geq c' \cdot t(n)$ for all $n \geq n'_0$.
Hence, if $t(n) \leq c \cdot g(n)$, then $g(n) \geq \frac{1}{c} \cdot t(n)$.
Therefore, the statement is true.

(b) $\Theta(\alpha g(n)) = \Theta(g(n))$, where $\alpha > 0$.

Proof:
By definition of Theta, $f(n) = \Theta(g(n))$ if $f(n)$ is both $O(g(n))$ and $\Omega(g(n))$.
Given $\alpha > 0$, $\alpha g(n)$ is simply a constant multiple of $g(n)$.
Constants do not affect the growth rate, thus $\Theta(\alpha g(n)) = \Theta(g(n))$.
Therefore, the statement is true.

(c) $\Theta(g(n)) = O(g(n)) \cap \Omega(g(n))$.

Proof:
By definition, $\Theta(g(n))$ means the function is bounded both above and below by $g(n)$ up to constant factors.
$O(g(n))$ defines an upper bound, and $\Omega(g(n))$ defines a lower bound.
Intersection of these two sets means the function is both bounded above and below by $g(n)$, which is the definition of $\Theta(g(n))$.
Therefore, the statement is true.

(d) For any two nonnegative functions $t(n)$ and $g(n)$ defined on the set of nonnegative integers, either $t(n) \in O(g(n))$ or $t(n) \in \Omega(g(n))$, or both.

Proof/Disproof:
Consider $t(n) = n$ and $g(n) = n$.
Both functions are equal, hence $t(n) \in O(g(n))$ and $t(n) \in \Omega(g(n))$.
Consider $t(n) = n$ and $g(n) = n^2$.
$t(n) \in O(g(n))$ since $n \leq n^2$.
Consider $t(n) = n^2$ and $g(n) = n$.
$t(n) \in \Omega(g(n))$ since $n^2 \geq n$.
Therefore, the statement is true as one of these conditions will always hold.

Problem 3

Determine the order of growth ($\Theta(\cdot)$) for each of the following functions. Show your work.

(a) $T(n) = \sum_{i=1}^{2n} i$

Sum of first $2n$ natural numbers:

$$\frac{2n(2n+1)}{2} = 2n^2 + n$$

Order of growth: $\Theta(n^2)$

(b) $T(n) = \sum_{i=1}^{n} \sum_{j=i}^{n} n$

Inner sum analysis:

$$\sum_{j=i}^{n} n = (n - i + 1)n$$

Outer sum analysis:

$$\sum_{i=1}^{n} (n - i + 1)n = n \sum_{i=1}^{n} (n - i + 1) = n \sum_{k=1}^{n} k = n \cdot \frac{n(n+1)}{2}$$

Order of growth: $\Theta(n^3)$

(c) $T(n) = \sum_{i=1}^{n} n^2$

Sum analysis:

$$n \cdot n^2 = n^3$$

Order of growth: $\Theta(n^3)$

(d) $T(n) = \sum_{i=1}^{n^2} i$

Sum of first n^2 natural numbers:

$$\frac{n^2(n^2+1)}{2} \approx n^4$$

Order of growth: $\Theta(n^4)$

(e) $T(n) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{\ell=1}^{n} \ell$

Innermost sum:

$$\sum_{\ell=1}^{n} \ell = \frac{n(n+1)}{2} \approx n^2$$

Next sum:

$$\sum_{k=1}^{n} n^2 = n^3$$

Next sum:

$$\sum_{j=1}^{n} n^3 = n^4$$

Outermost sum:

$$\sum_{i=1}^{n} n^4 = n^5$$

Order of growth: $\Theta(n^5)$