

# Algorithms: Design and Analysis - CS 412

## Problem Set 01: Asymptotic Analysis

1. Let

$$p(n) = \sum_{i=0}^d a_i n^i$$

where  $a_d > 0$ , be a degree- $d$  polynomial in  $n$  and let  $k$  be a constant. Use the definition of the asymptotic notations to prove the following properties:

- (a) If  $k \geq d$ , then  $p(n) = O(n^k)$ .
- (b) If  $k \leq d$ , then  $p(n) = \Omega(n^k)$ .
- (c) If  $k = d$ , then  $p(n) = \Theta(n^k)$ .
- (d) If  $k > d$ , then  $p(n) = o(n^k)$ .
- (e) If  $k < d$ , then  $p(n) = \omega(n^k)$ .

### Solution:

- (a) If  $k \geq d$ , then  $p(n) = O(n^k)$ .

Definition of Big-Oh:  $f(n) = O(g(n))$  if there exists positive constants  $c$  and  $n_0$  such that  $0 \leq f(n) \leq c \cdot g(n) \quad \forall n \geq n_0$

*Proof.* Choose  $c = \sum_{i=0}^d |a_i|$  and  $n_0 = 1$ . Then  $\forall n \geq n_0$ :

$$p(n) = \sum_{i=0}^d a_i n^i \leq \sum_{i=0}^d |a_i| n^d \leq \left( \sum_{i=0}^d |a_i| \right) n^d = c n^d$$

Since  $k \geq d$ ,  $n^d \leq n^k \quad \forall n \geq 1$ , thus  $p(n) = O(n^k)$  □

- (b) If  $k \leq d$ , then  $p(n) = \Omega(n^k)$ .

Definition of Big-Omega:  $f(n) = \Omega(g(n))$  if there exists positive constants  $c$  and  $n_0$  such that  $0 \leq c \cdot g(n) \leq f(n) \quad \forall n \geq n_0$

*Proof.* Choose  $c = a_d$  and  $n_0 = 1$ . Then  $\forall n \geq n_0$ :

$$p(n) = \sum_{i=0}^d a_i n^i \geq a_d n^d \geq a_d n^k = cn^k$$

Since  $a_d > 0$  and  $k \leq d, n^d \geq n^k \quad \forall n$ , thus  $cn^k$  is a lower bound for  $p(n)$ , and  $p(n) = \Omega(n^k)$ .  $\square$

(c) If  $k = d$ , then  $p(n) = \Theta(n^k)$ .

Definition of Big-Theta:  $f(n) = \Theta(g(n))$  if there exists positive constants  $c_1, c_2$  and  $n_0$  such that  $0 \leq c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n) \quad \forall n \geq n_0$ . Or in other words,  $f(n) = \Theta(g(n))$  if  $f(n) = O(g(n))$  and  $f(n) = \Omega(g(n))$ .

*Proof.* From parts (a) and (b), we have shown that if  $k \geq d$ , then  $p(n) = O(n^k)$  and if  $k \leq d$ , then  $p(n) = \Omega(n^k)$ . When  $k = d$ , both conditions are satisfied, which means  $p(n)$  is both upper and lower bounded by  $n^k$ , hence is both  $O(n^k)$  and  $\Omega(n^k)$ , and therefore  $p(n) = \Theta(n^k)$ .  $\square$

**2.** Indicate for each pair of expressions  $(A, B)$  in the table below, whether A is  $O, o, \Omega, \omega$ , or  $\Theta$  of B. Assume that  $k \geq 1, \epsilon > 0$ , and  $c > 1$  are constants. Write your answer in the form of the table with “yes” or “no” written in each box.

	$A$	$B$	$O$	$o$	$\Omega$	$\omega$	$\Theta$
<b>a.</b>	$\lg^k n$	$n^\epsilon$					
<b>b.</b>	$n^k$	$c^n$					
<b>c.</b>	$\sqrt{n}$	$n^{\sin n}$					
<b>d.</b>	$2^n$	$2^{n/2}$					
<b>e.</b>	$n^{\lg c}$	$c^{\lg n}$					
<b>f.</b>	$\lg(n!)$	$\lg(n^n)$					

**3.** Let  $f(n)$  and  $g(n)$  be asymptotically positive functions. Prove or disprove each of the following conjectures.

- (a)  $f(n) = O(g(n))$  implies  $g(n) = O(f(n))$ .
- (b)  $f(n) + g(n) = \Theta(\min\{f(n), g(n)\})$ .
- (c)  $f(n) = O(g(n))$  implies  $\lg f(n) = O(\lg g(n))$ , where  $\lg g(n) \geq 1$  and  $f(n) \geq 1$  for all sufficiently large  $n$ .
- (d)  $f(n) = O(g(n))$  implies  $2^{f(n)} = O(2^{g(n)})$
- (e)  $f(n) = O((f(n))^2)$ .

(f)  $f(n) = O(g(n))$  implies  $g(n) = \Omega(f(n))$ .

(g)  $f(n) = \Theta(f(\frac{n}{2}))$

(h)  $f(n) + o(f(n)) = \Theta(f(n))$

4. Let  $f(n)$  and  $g(n)$  be asymptotically positive functions. Prove the following identities.

(a)  $\Theta(\Theta(f(n))) = \Theta(f(n))$

(b)  $\Theta(f(n)) + O(f(n)) = \Theta(f(n))$

(c)  $\Theta(f(n)) + \Theta(g(n)) = \Theta(f(n) + g(n))$

(d)  $\Theta(f(n)) \cdot \Theta(g(n)) = \Theta(f(n) \cdot g(n))$