CSCE 750, Fall 2021, Homework 1
Solutions and Hints to Selected Problems

September 1, 2021

Exercises

Page 22 Exercise 2.1-2

Answer: Identical to the pseudocode given on page 18 except line 5 is now

while $i > 0$ and $A[i] < key$

Pages 41–42 Problem 2-4

Answer:

b. The most inversion occur when the array is in descending order (why?) with exactly $n(n-1)/2$ inversions (why?).

c. If v is the number of inversions on n elements, then the running time of InsertionSort is $\Theta(n+v)$. (Why?)

d. Idea: In MergeSort, you can maintain a global count of inversions, initialized to 0. When merging, every time you take from the right list, increment the count by the current length of the remaining left list. (Why does this work?)

Pages 52–53 Exercises 3.1-1, 3.1-2 (note that b need not be an integer, and a need not be positive), 3.1-3, 3.1-4

Answer:

3.1-1 For all sufficient large n, we have $f(n) \geq 0$ and $g(n) \geq 0$ by assumption. Also note that $f(n) + g(n) = \max(f(n), g(n)) + \min(f(n), g(n))$. Thus for all sufficiently large n,

$$\max(f(n), g(n)) = f(n) + g(n) - \min(f(n), g(n)) \leq f(n) + g(n) \in O(f(n) + g(n))$$

(with constant 1). For the lower bound, we have

$$f(n) + g(n) = \max(f(n), g(n)) + \min(f(n), g(n)) \leq 2\max(f(n), g(n)) ,$$

and so $\max(f(n), g(n)) \geq (f(n) + g(n))/2 \in \Omega(f(n) + g(n))$ (with constant 1/2). Thus by definition, $\max(f(n), g(n)) \in \Theta(f(n) + g(n))$.

3.1-2 For all $n \geq 2|a|$, we have

$$(1/2)^b n^b = (n/2)^b \leq (n-|a|)^b \leq (n+a)^b \leq (n+|a|)^b \leq (3n/2)^b = (3/2)^b n^b \ .$$

Thus $(n+a)^b \in \Theta(n^b)$ via the constants $(1/2)^b$ and $(3/2)^b$. [We repeatedly use the fact that, because $b > 0$, the function $f(x) := x^b$ is monotone increasing on the positive reals.]

3.1-4 Yes. No.

(not in the textbook) Give an example of two functions $f(n)$ and $g(n)$, both eventually positive, for which $f(n) \notin O(g(n))$ and $g(n) \notin O(f(n))$. Prove that your answer is correct. [This shows that the eventually positive functions cannot be totally ordered by asymptotic growth rate.]

Answer: For natural numbers n, define

$$f(n) := \begin{cases} n & \text{if } n \text{ is odd,} \\ 1 & \text{if } n \text{ is even,} \end{cases}$$

and define

$$g(n) := n + 1 - f(n) = \begin{cases} n & \text{if } n \text{ is even,} \\ 1 & \text{if } n \text{ is odd.} \end{cases}$$

(Proof of correctness omitted.)

Page 1149 Exercises A.1-1, A.1-3 (Hint: Start with Equation A.8, then differentiate both sides.), A.1-7

Answer:

A.1-1 n^2 (why?)

A.1-7

$$\prod_{k=1}^{n} 2 \cdot 4^k = 2^n \cdot \prod_k 4^k = 2^n \cdot 4^{\sum_{k=1}^n k} = 2^n \cdot 4^{n(n+1)/2} = 2^n \cdot (2^2)^{n(n+1)/2}$$
$$= 2^n \cdot 2^{n(n+1)} = 2^{n+n(n+1)} = 2^{n(n+2)}$$

Page 1156 Exercises A.2-1, A.2-2, A.2-4

Answer:

A.2-1 Hint: There are more than one way of doing this. You can first bound the sum from above by an integral (Equation (A.12)) after separating the first term in the sum. You could instead chop the sum into blocks B_i, where the ith block B_i has 2^i successive terms, then bound each block.

A.2-2 Using the inequalities $\lfloor x \rfloor \leq x$ and $\lceil x \rceil < x + 1$, we have, for all sufficiently large n,

$$\sum_{k=0}^{\lfloor \lg n \rfloor} \left\lceil n/2^k \right\rceil < \sum_{k=0}^{\lfloor \lg n \rfloor} (n/2^k + 1) = n \cdot \sum_{k=0}^{\lfloor \lg n \rfloor} (1/2)^k + \sum_{k=0}^{\lfloor \lg n \rfloor} 1 < n \cdot \sum_{k=0}^{\infty} (1/2)^k + \lfloor \lg n \rfloor + 1$$
$$= n \cdot \frac{1}{1 - 1/2} + \lfloor \lg n \rfloor + 1 \leq 2n + \lg n + 1 \in O(n) \ .$$

This bound is tight, because the $k = 0$ term in the sum is already n.

A.2-4 The function $f(x) := x^3$ is monotonically increasing, so by Equation (A.11),

$$\int_0^n x^3\,dx \le \sum_{k=1}^{n} k^3 \le \int_1^{n+1} x^3\,dx\ .$$

Evaluating the integrals, we get

$$\frac{n^4}{4} \le \sum_{k=1}^{n} k^3 \le \frac{1}{4}((n+1)^4 - 1) = \frac{n^4}{4} + n^3 + \frac{3n^2}{2} + n = \frac{n^4}{4} + O(n^3)\ .$$