

Spencer Shimko
CMSC 441
HW 1

1. Exercise 2.3-7, page 37. (Hint: you can sort in $O(n \lg n)$ time, so you might as well start with that.) Describe a $\Theta(n \lg n)$ -time algorithm that, given a set S of n integers and another integer x , determines whether or not there exist two elements in S whose sum is exactly x .

Inputs: set(s); integer x;

```
S <- MergeSort(S)
for each z in S
    if BinarySearch(S, (x-z)) returns non-zero
        return true
    end-if
end-for
return false
```

2. Give simple examples of functions $f(n)$ having the following properties:

a) $f(n) = \Theta(f(n+1))$
 $f(n) = k$ for some arbitrary constant k

b) $f(n) = o(f(n+1))$
 $f(n) = n - 1$

c) $f(n) = \omega(f(n+1))$
 $f(n) = n$

3. Problem 3-4, page 58. 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 $g(n) = O(f(n))$.
This is not true. Simply put: $n = O(n^2)$ but $n^2 \neq O(n)$

b) $f(n) + g(n) = \Theta(\min(f(n), g(n)))$.
Not true. By example: $2^n + 1 \neq \Theta(\min(2^n, 1)) = \Theta(1)$

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 .
True. By definition: $f(n) \leq cg(n) \rightarrow \lg(f(n)) \leq \lg(CG(n))$
Also note: $\lg(CG(n)) = \lg(c) + \lg(g(n))$ also by definition. Now we have to prove that $(\lg(c) + \lg(g(n))) \leq x \lg(g(n))$ for some constant x .
 $\lg(g(n)) = \lg(c) + \lg(g(n))$

$$b = (\lg(c)/\lg(g(n))) + (\lg(g(n))/\lg(g(n)))$$

$$b = (\lg(c)/\lg(g(n))) + 1 \leq \lg(c) + 1 \text{ since } \lg(g) \geq 1$$

d) $f(n) = O(g(n))$ implies $2^{f(n)} = O(2^{g(n)})$.

This does not hold true. Example:

$$f(n) = 2n; g(n) = n;$$

$$2n = O(2^n) \text{ but } 2^{(2n)} \neq O(2^n)$$

e) $f(n) = O((f(n))^2)$.

This is true for all nondecreasing $f(n)$. i.e. $f(n) > 1$. If it is less than 1, $O((f(n))^2)$ is actually smaller than any actual upper bounds.

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

This holds true. The definitions tell us that:

$$f(n) \leq cg(n)$$

$$1/c f(n) \leq g(n)$$

g) $f(n) = \Theta(f(n/2))$.

No. Easily:

$$\text{Let } f(n) = 2^n$$

$$2^n \leq c(2^{n/2}) \text{ (not true)}$$

It will fail for any sufficiently large case of n in exponentiation.

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

This holds true since $f(n) + o(f(n)) = \Theta(\max(f(n), o(f(n))))$

4. Show that the sum, for k running from 1 to n , of $(\lg k) / k$, is $\Theta(\lg^2 k)$, in two ways:

a) by integration, and

b) by bounding the individual terms (this is tricky!)
(Hint: you might use the very useful result (A.7)).