

COP 3503 Honors – Homework 1 (Collaborative)

Due Date: September 12, 2014

1. Problem 3-3(a) in Cormen et al. (pg. 61-62). Do not rank the iterated log functions. Make sure to justify the rankings. Hint: Using limits and L'Hospital's rule can be useful here.
2. You are given the following function that takes a nonnegative integer as an argument and returns an integer.

```
F(i)
1  if (i = 0)
2    return 1
3  return (2 * F(i - 1))
```

- (a) Analyze the running time of the function F. State your assumptions clearly.
 - (b) Give a function FNEW(i) that always returns the same value as F(i) but has lower time complexity. Analyze its running time.
3. Solve the following recurrences and justify your answers
 - (a) Use the iteration method or a recursion tree
 - $T(n) = 2T(n - 2) + n$
 - $T(n) = 3T(\sqrt{n}) + \lg n$
 - (b) Use the substitution method
 - $T(n) = 2T(\sqrt{n}) + n, T(2) = 2$ (Guess: $T(n) = \Theta(n)$)
 - $T(n) = 2T(\sqrt{n}) + \lg n, T(1) = 1$ (Guess: $T(n) = \Theta(\lg n \lg \lg n)$)
 - (c) Use the Master Method (find tight asymptotic bounds)
 - $T(n) = 2T(n/2) + n^3$
 - $T(n) = T(9n/10) + n$
 - $T(n) = 16T(n/4) + n^2$
 - $T(n) = 7T(n/2) + n^2$
 4. Describe a $\Theta(n \lg n)$ -time algorithm that, given a set S of n integers and another x , determine whether or not there exist two elements in S whose sum is exactly x .