Assignment 1 – Ryan Harding

1. Prove that O(Log10n) = O(Log5n)
   1. Base Case:
      1. Algorithm A asymptomatic complexity is O(Log10n)
      2. Algorithm B asymptomatic complexity is O(Log5n)
   2. Given:
   3. Solution
      1. - A constant factor does not affect asymptomatic complexity per the mathematical definition of Big O
      2. Therefore
2. Analyze the running time of the recursive procedure:
   1. Function mystery(n) {

if(n <= 1)

return;

else{

mystery(n / 3); //Makes one recursive call

Var count = 0;

for(var i = 0; i < n; i++) { //Iterates n times

count=count+ 1;

}

mystery(n / 3); //Makes second recursive call

}

}

* 1. Solution:
     1. Recurrence Relation:
        1. 2 calls made to mystery(n / 3)
        2. One loop running n times
        3. T(n) = 2T (n/3) + n
     2. Apply Master’s Theorem:
        1. T(n) = AT(n/b) + f(n)
        2. A = 2, B = 3
        3. Find
        4. When
        5. Then T(n) = O(n)

1. Sort the list 2,4,10,6,-1,3 using Insertion Sort:
   1. COME BACK TO THIS!