Homework 1

Problem 1.

Collaboration: None

Step 1. Group into classes

1. Logarithmic: (log(3n))^3, log(2n^2), 2log(n), (log(n))^¼
2. Linear: 4^log(n)
3. Polynomial: 4n+16n^6/5, n^3/2, 4n^2
4. Extra Log: n^2log(3n)
5. Double Log: 2log(log(2n)), log(log(2n))^2
6. Exponential: 2^1.5n, 5^n/2, 5^n-3, 5^n, 0.99^n
7. Factorial: 4(n!), (7n/8)!

Step 2. Check within each class to see if equivalent

1. Logarithmic:
   1. None of the Logarithmic functions are equivalent
2. Polynomial:
   1. None of the polynomial functions are equivalent
3. Double Log:
   1. log(log(2n))^2 2log(log(2n)) by the rules of log functions so therefore 2log(log(2n)) and log(log(2n))^2 are equivalent
4. Exponential:
   1. None of the Exponential functions are equivalent
5. Factorial:

Step 3. Find Bounds for functions

1. Logarithmic:
2. Linear:
3. Polynomial:
4. Extra Log:
5. Double Log:
6. Exponential:
7. Factorial:

Step 4. Arrange classes

Problem 2.

Collaboration: None

1. Part 1
2. Part 2
   1. Base Case h = 0
   2. Suppose h = k
   3. If h = k+1
      1. Substitute
      2. The hypothesis holds because for all possible heights it will be less than

Problem 3.

Collaboration: None

Code Segment 1: The first time it is called it runs i=n and j = 1, and it runs 2(n+1) + 5 times, then i=n/2 and j=3 and it runs 2(n/2 + 3)+5 times. Then i = n/4 and j = 9 and it runs 2(n/4 + 9) + 5 times etc…

So it gets called times

Solving this we get 2n(2-(½)^log(n)) + 8/3(3^log(n)-1 + 5log(n)

Which gives us O(3^log(n))

Code Segment 2: The first time j = 1 and k = 0 results in no calls. Once j = 8, k = 1 and r=1 so we get one call. And while j = 9 to j = 15 we get a call each time. From j = 16 to j = 31, k=2 and so we run it 4 times each. And from j = 32 to j = 63 we call 9 times each

This creates a series (0^3)(8) + (1^3)(8)+(2^3)(8) + … n/8 times

Simplifying we get (8/6) \* ((n/8)((n/8) + 1)(2(n/8) + 1))

Which gives us O(n^3)