ECSE210L: Design and Analysis of Algoithms

Tutorial 2 (Week 2: January, 13 - 17, 2020)

Instructors: ***Shakti Sharma and Raghunath Reddy M***

1. Prove or disprove the following.
   a) $2^{n+1} = O(2^n)$
   b) $2^{2n} = O(2^n)$
2. Take the following list of functions and arrange them in ascending order of growth rate. That is, if function $g(n)$ immediately follows function $f(n)$ in your order, then it should be the case that $f(n) = O(g(n))$
   - $f_1(n) = 10^n$
   - $f_2(n) = n^{1/3}$
   - $f_3(n) = n^n$
   - $f_4(n) = \log_2 n$
   - $f_5(n) = 2^{\sqrt{\log_2 n}}$
3. Take the following list of functions and arrange them in ascending order of growth rate.
   - $f_1(n) = n^{0.999999} \log n$
   - $f_2(n) = 100000000n$
   - $f_3(n) = (1.000001)^n$
   - $f_4(n) = n^2$
   - $f_5(n) = n^2 \log n$
4. Take the following list of functions and arrange them in ascending order of growth rate.
   - $f_1(n) = 2^{\sqrt{\log n}}$
   - $f_2(n) = n^{\frac{4}{3}}$
   - $f_3(n) = n(\log n)^3$
   - $f_4(n) = n^{\log n}$
   - $f_5(n) = 2^{2^n}$
   - $f_6(n) = 2^{n^2}$
5. Let $f(n)$ and $g(n)$ be two functions such that n is a non-negative real number, and suppose that $f(n) = O(g(n))$. Show that $g(n) = \Omega(n)$.