Christian Rodriguez

Algorithms HW 4

2. Prove that if f (n) is a polynomial of the form
Xd
i=1
ainxi , for some coe_-
cients a1; a2; : : : ; ad and exponents x1; x2; : : : ; xd , then f (n) = _(nmaxfx1;x2;:::;xdg).
Hint : you may use any property of Big-Oh notation listed in the slides.
You may wish to use induction for this problem.
3. (Bonus) Prove that 2n = (nk) for all integers k _ 1.

Proof:

1. Use the formal definition of Big-Oh to prove that if f(n)=O(g(n)), then f(n)+g(n)=O(g(n))
   - Proof: Since f(n)=O(g(n)), there exists positive constants c and n_0 ($\exists c>0, n_0$) such thatf(n)<=c*g(n) for sufficiently large n ($\forall n>=n_0$). We add both sides by g(n), yielding f(n)+g(n)=c*g(n)+g(n). f(n)+g(n)<=c*g(n)+g(n)=(c+1)*g(n)=c_1*g(n) where c_1=c+1 and $\forall n>=n_0$. Therefore, using the property of coefficients, f(n)+g(n)=O(g(n)).
2. Prove that if f(n) is a polynomial of the form

$$\sum_{i=0}^{k+1} a_i n\text{^}(x_i)$$

   for some coefficients a,a,…,a and exponents and exponents x,x,…,x, then f(n) = $\Theta(n^{\max\{x,x,\ldots,x_d\}})$.
   - Show: f(n) = $\Theta(n^{\max\{x,\ldots,x\}})$
   - Do induction of # of terms in polynomial in polynomials.
   - (Base case: d=1)
     - $f(n)=a_1n^{x_1}= \Theta(n^{x_1})= \Theta(n^{\max_{(x,d)}})$
     - aka: $a_1n^{x_1} <= a_1n^{x_1} <= a_1n^{x_1}, \forall n >= 1$
     - $a_1n^{x_1} = \Theta(n^x)$
   - Inductive Step:
     - Suppose any polynomial of k terms is $\Theta(n^x)$, where x is the exponent, and consider
   - f(n) =$\sum_{i=0}^{k+1} a_i n\text{^}(x_i)$
   - $=g(n) + a_{k+1}n^{x_{k+1}}$
   - $=g(n)=a_1n^{x_1} + \ldots + a_kn^{x_k}$
   - By Inductive Hypothesis, g(n)= $\Theta(n^{x\ldots})$, x…=max{$x_1,\ldots,x_k$}
   - F(n)= $\Theta(n^x)+a_k$
   - $\Theta(n^x)+ \Theta(n^{x_{k+1}})$
   - $\Theta(n^x + n^{x_{k+1}})$
   - Split into base cases:
   - Case 1:
     - $x >= x_{k+1}$
     - $\Theta(n^x + n^{x_{k+1}})= \Theta(n^x)$
     - $n^{x_{k+1}}=O(n^x)$

- $x=\max\{x_1,\ldots,x_k\}$
- $x=\max\{x_1,\ldots,x\}$
- because $x>=x_{k+1}$

- Case 2:
  - $x < x_{k+1}$
  - $n^x=O(n^{x_{k+1}})$
  - $\Theta(n^x + n^{x_{k+1}}) = \Theta(n^{x_{k+1}})$
  - $x_{k+1} = \max\{x_1,\ldots,x_k,x_{k+1}\}$
  - because $x_{k+1} > \max\{x_1,\ldots,x_k,x_k\}$

3. Prove that $2^n=\Omega(n^k)$ for all integers $k>=1$.
   - $f(n)= \Omega(g(n))$ if and only if there exist positive constants c and n_0 such that $f(n) >=c*g(n)$ for all $n>=n_0$.
   - Since $\lim_{n->infinity}(n\hat{}k)/(2\hat{}n)$=0 and 0<(n^k)/(2^n)<1 for sufficiently large n, and $0<n^k<=2^n$ for all n. This matches the definition of $2^n=\Omega(n^k)$, with c=1.
     - I am citing the following website as a source. It uses Big Oh notation but I believe that it is a close enough solution to the given problem.
     - www.math.stackexchange.com/questions/367767/how-to-prove-that-nk-o2n