CS 310

Assignment 0907

Christian Thomas

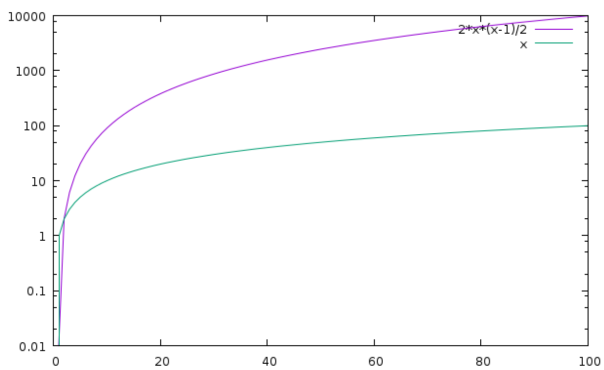
Problem 1. Use the definitions to prove or disprove the statement $2n(n-1)/2 \in \Theta(n)$, and illustrate this graphically.

Answer: The definition requires us to find c and n_0 so that

$$\frac{2n(n-1)}{2} \geq cn \text{ when } n \geq n_0$$

To prove that $2n(n-1)/2 \in \Theta(n)$ we first must prove $2n(n-1)/2 \in O(n)$ and $2n(n-1)/2 \in \Omega(n)$. The book states that $g(n) \neq O(f(n))$ if the ratio $g(n)/f(n)$ is getting arbitrarily large and no constant c could make definition work. If we take $(2n(n-1)/2)/(n)$ you get $n-1$ which will grow arbitrarily large, no matter the constant.

This is graphically illustrated by the following plot that shows $2n(n-1)/2$ along with the standard function n^{n^n} scaled by the constant coefficient $c = 22$.



Problem 2. Using the definitions, either prove the following assertion, or disprove it with a specific counterexample:

$$\text{if } t(n) \in O(g(n)) \text{ then } g(n) \in \Omega(t(n))$$

Answer: If the statement is true, then there exist c_1, c_2, and n_0 such that $g(n) \geq c_1 * t(n)$ for Big O and $t(n) \leq c_2 * g(n)$ for Big Omega. The next, Big O turns into $g(n)/t(n) \geq c_1$ and Big Omega turns into $t(n)/g(n) \leq c_2$. Lastly, you then get $t(n) \leq (1/c_1) * g(n)$ for Big O and $g(n) \geq (1/c_2) * t(n)$ for Big Omega which makes sense for any $n \geq n_0$ QED.

Problem 3. For each of the following pairs, state and briefly justify (but you dont have to rigorously prove) whether $f(n) \in O(g(n))$, $f(n) \in \Omega(g(n))$, or $f(n) \in \Theta(g(n))$.

1. $f(n) = n^{1/2}$ $g(n) = n^{2/3}$ **Answer:** $n^{1/2} \in \Theta(n^{2/3})$ is true because both functions are polynomials functions with degrees between 0 and 1.

2. Answer: The constant 10 in front of log(n) is practically irrelevant at insanely large values making it essentially $log(n)$ vs. $log(n^2)$ since it is well know that $n \leq O(n^2)$ is true, then we know that $10log(n) \in O(log(n^2))$ is also true.

3. Answer: In this question it is a polynomial function vs. a logarithmic function. It is well known that polynomial functions tend to be much slower than logarithmic, so $n^{1/2} \in \Omega((log(n)^3))$

4. Answer: Addition becomes meaningless when headed toward infinity, which means we are essentially comparing n and $nlog(n)$. n and n obviously grow at same rate, so if you multiply n by $log(n)$ that would make $n \leq nlog(n)$. Therefore, $n + log(n) \in O(nlog(n))$