Homework 1 CS41, Spring 2023 Nina Zhuo

1. Fore each of the following pairs of functions, indicate whether $f = o(g)$, $f = \omega(g)$, $f = \theta(g)$, or none of these hold. You don't need to prove your answers.

   (a) for $f = n - 100$ and $g = n - 200$, $f = \theta(g)$

   (b) for $f = n^{1/2}$ and $g = n^{2/3}$, $f = o(g)$

   (c) for $f = 100n + log(n)$ and $g = n + (logn)^2$, $f = \theta(g)$

   (d) for $f = nlogn$ and $g = 10n(log10n)$, $f = \theta(g)$

   (e) for $f = log2n$ and $g = log3n$, $f = \theta(g)$

   (f) for $f = 10logn$ and $g = logn^2$, $f = \theta(g)$

   (g) for $f = n \cdot 2^n$ and $g = 3^n$, $f = o(g)$

   (h) for $f = 2^n$ and $g = 2^{n+1}$, $f = \theta(g)$

   (i) for $f = n^2/logn$ and $g = n(logn)^2$, $f = \omega(g)$

   (j) for $f = n^{1.01}$ and $g = n(logn)^2$, $f = o(g)$

   (k) for $f = n^{0.1}$ and $g = (logn)^{10}$, $f = o(g)$

   (l) for $f = \sqrt{n}$ and $g = (logn)^3$, $f = \omega(g)$

   (m) for $f = \sqrt{n}$ and $g = 5^{logn}$, $f = o(g)$

   (n) for $f = n^{1+(-1)^n}$ and $g = n$, none of these hold

   (o) for $f = n!$ and $g = 2^n$, $f = \omega(g)$

   (p) for $f = (logn)^{logn}$ and $g = 2^{logn^2}$, $f = o(g)$

2. For each of the following statements, either prove (using the formal definition of big O) that it is true for all functions f, g, and h from positive integers to positive reals, or give a counterexample.

Big-O : $f = O(g)$ if $f \leq c \cdot g$ for all n when $n_0 \geq 1$ and c is a positive, non-zero integer

(a) If $f = O(g)$ and $g = O(h)$, then $f = O(h)$.

Proof. Because $f = O(g)$ and $g = O(h)$, we know that $f \leq c_1 \cdot g$ and $g \leq c_2 \cdot h$ by definition of big-O. Assume c_1 and c_2 are positive, non-zero integers.

We can rewrite this as $f \leq c_1 \cdot g \leq c_2 \cdot h$ which simplifies to $f \leq c \cdot h$

Therefore, $f = O(h)$ by definition of big-O.

$\square$

(b) If $f = O(h)$ and $g = O(h)$, then $f + g = O(h)$.

Proof. Because $f = O(h)$ and $g = O(h)$, we know that $f \leq c_1 \cdot h$ and $g \leq c_2 \cdot h$. Suppose $g \leq f$ such that

$$g \leq f \leq c \cdot h \qquad \text{and} \qquad f + g \leq 2f$$

As such, it must also be true that

$$f + g \leq 2f \leq 2 \cdot c \cdot h$$

Since there exists some constant $k = 2c$ such that $f + g \leq k \cdot h$, we know that if $f = O(h)$ and $g = O(h)$, then $f + g = O(h)$ by definition of big-O. $\square$

(c) If $f = O(h)$ and $g = O(h)$, then $f \cdot g = O(h)$.

Because $f = O(h)$ and $g = O(h)$, we know that $f \leq c \cdot h$ and $g \leq c \cdot h$. By definition of big-O, $f \cdot g = O(h)$ is the same as $f \cdot g \leq c \cdot h$. However, in the case that $f = g = n$, we get

$$n^2 \leq c \cdot n$$

Because n^2 increases exponentially while $c \cdot n$ increases linearly by a factor of c, this inequality will not hold for all values of n such that $n \geq 1$. Therefore, $f \cdot g \neq O(h)$.

(d) If $f(n) = O(g(n))$, then $log(f(n)) = O(log(g(n)))$.

Proof. Because $f(n) = O(g(n))$, we know that $f \leq c \cdot g$.

Let us assume the maximum value of f and the minimum value of g such that $f = g = n$. Thus, we can rewrite the expression as:

$$log(n) \leq c \cdot log(n)$$

We know this inequality will hold true for all f and g such that $f \leq c \cdot g$ because the above assumes extreme cases of f and g. Therefore, $log(f(n)) = O(log(g(n))$ when $f(n) = O(g(n))$.

$\square$

(e) If $f(n) = O(g(n))$, then $2^{f(n)} \neq O(2^{g(n)})$.

Because $f(n) = O(g(n))$, we know that $f \leq c \cdot g$. In the case that $f = g = 3$, we get:

$$2^3 \leq c \cdot 2^3 \qquad \text{which is the same as} \qquad 8 \leq c \cdot 8$$

Assuming that c is some positive, non-negative integer, the inequality holds true, so when $f(n) = O(g(n))$, $2^{f(n)} = O(2^{g(n)})$.

(f) $f(n) = O(f(n/2))$ In the case that $f = 6^n$, we get

$$6^n \leq c \cdot \frac{6^n}{2}$$

While there may exist some c that would make this inequality true for some values of n, we find that there is no one constant c that holds true as n continues to increase. Therefore, $f(n) \neq O(f(n/2))$