Learning Objectives

By the end of this worksheet, you will:

- Prove statements using the definition of Big-Oh and its negation.
- Represent constant functions in Big-Oh expressions.
- Understand and use the definition of Omega and Theta to compare functions.

For your reference, here is the formal definition of Big-Oh:

$$g \in \mathcal{O}(f) : \ \exists c, n_0 \in \mathbb{R}^+, \ \forall n \in \mathbb{N}, \ n \geq n_0 \Rightarrow g(n) \leq cf(n)$$

1. **Constant functions**. One type of function we haven't talked about yet is the *constant function*, like $f(n) = 1$ or $f(n) = 100$. This type of function will play an important role in our analysis of running time next week, so for now let's get comfortable with the notation.

   (a) Let $g : \mathbb{N} \to \mathbb{R}^{\geq 0}$. Show how to express the statement $g \in \mathcal{O}(1)$ by expanding the definition of Big-Oh.[1]

   (b) Prove that $100 + \dfrac{77}{n+1} \in \mathcal{O}(1)$. (We add 1 in the denominator to ensure this function is defined when $n = 0$.)
   Note: this proof isn't too mathematically complex; treat this as another exercise in making sure you understand the definition of Big-Oh!
   Hint: one algebraic property you can use is that $\forall x, y \in \mathbb{R}^+, \ x \geq y \Rightarrow \dfrac{1}{x} \leq \dfrac{1}{y}$.

[1] Remember that we often abbreviate Big-Oh expressions to just show the function bodies. "$\mathcal{O}(1)$" is really shorthand for "$\mathcal{O}(f)$, where f is the constant function $f(n) = 1$."

2. **Omega**. We can think of Big-Oh notation as describing and *upper bound* on the rate of growth of a function. Saying "$g \in \mathcal{O}(f)$" is like saying "g grows at most as fast as f." As you might expect, sometimes we care just as much about a *lower bound* on the rate of growth. For this, we have the symbol Ω (the Greek letter Omega), which is defined analogously to Big-Oh:

$$g \in \Omega(f) : \ \exists c, n_0 \in \mathbb{R}^+, \ \forall n \in \mathbb{N}, \ n \geq n_0 \Rightarrow g(n) \geq cf(n)$$

Using this definition, prove that for all $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$, if $g \in \mathcal{O}(f)$, then $f \in \Omega(g)$.

3. **Theta**. We have now seen Big-Oh, which is used to specify upper bounds on rates of growth, and Omega, which is used to specify lower bounds on rates of growth. However, both of these symbols are limited in the same way as inequalities on numbers. "$2 \leq 10^{10}$" is a true statement, but not very insightful; similarly, "$n+1 \in \mathcal{O}(n^{10})$" and "$2^n + n^2 \in \Omega(n)$" are both true, but not very precise.

   Our final piece of notation is the symbol Θ (the Greek letter Theta), which can be defined in rather simple terms:

   $$g \in \Theta(f): \; g \in \mathcal{O}(f) \wedge g \in \Omega(f)$$

   Or equivalently,

   $$g \in \Theta(f): \; \exists c_1, c_2, n_0 \in \mathbb{R}^+, \; \forall n \in \mathbb{N}, \; n \geq n_0 \Rightarrow c_1 f(n) \leq g(n) \leq c_2 f(n)$$

   When we write $g \in \mathcal{O}(f)$, what we mean is "g grows at most as quickly as f *and* g grows at least as quickly as f" – in other words, that f and g have the *same* rate of growth. We call f a *tight bound* on g, since g is essentially squeezed between constant multiples of f.

   Prove that for all functions $g : \mathbb{N} \to \mathbb{R}^{\geq 0}$, and all numbers $a \in \mathbb{R}^{\geq 0}$, if $g \in \Omega(1)$, then $a + g \in \Theta(g)$.[2]

   [Or in other words, for such functions g, shifting them by a constant amount does not change their "Theta" bound.]

[2] Here we use $a + g$ to denote the function g_1 defined as $g_1(n) = a + g(n)$ for all $n \in \mathbb{N}$.

4. **Negating Big-Oh**. So far, we have only looked at proving that a function *is* Big-Oh of another function. In this question, we'll investigate what it means to show that a function *isn't* Big-Oh of another.

   (a) Express the statement $g \notin \mathcal{O}(f)$ in predicate logic, using the expanded definition of Big-Oh. (As usual, simplify so that all negations are pushed as far inside as possible.)

   (b) Prove that for all positive real numbers a and b, if $a > b$ then $n^a \notin \mathcal{O}(n^b)$.
   Hint: for all positive real numbers x and y, $x > y \Leftrightarrow \log x > \log y$.