Learning Objectives

By the end of this worksheet, you will:

- Prove and disprove statements using the definition of Big-Oh.
- Investigate properties of Big-Oh of some common functions.

Note: in Big-Oh expressions, it will be convenient to just write down the “body” of the functions rather than defining named functions all the time. We’ll always use the variable \( n \) to represent the function input, and so when we write “\( n \in O(n^2) \),” we really mean “the functions defined as \( f(n) = n \) and \( g(n) = n^2 \) satisfy \( f \in O(g) \).”

As a reminder, here is the formal definition of what “\( g \) is Big-Oh of \( f \)” means:

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

1. Comparing polynomials. Our first step in comparing different types of functions is looking at different powers of \( n \). Consider the following statement, which generalizes the idea that \( n \in O(n^2) \):

\[
\forall a, b \in \mathbb{R}^+, a \leq b \Rightarrow n^a \in O(n^b)
\]

(a) Rewrite the above statement, but with the definition of Big-Oh expanded.

Solution

\[
\forall a, b \in \mathbb{R}^+, a \leq b \Rightarrow (\exists c, n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow n^a \leq cn^b)
\]

(b) Prove the above statement. Hint: you can actually pick \( c \) and \( n_0 \) to both be 1, and have the proof work.

Solution

Proof. Let \( a, b \in \mathbb{R} \), and assume \( a \leq b \). Let \( c = 1 \) and \( n_0 = 1 \). Let \( n \in \mathbb{N} \), and assume that \( n \geq n_0 \). We want to prove that \( n^a \leq n^b \).

We can start with our assumption that \( a \leq b \) and calculate:

\[
\begin{align*}
  a & \leq b \\
  n^a & \leq n^b \quad \text{(since } n \geq 1) \\
  n^a & \leq cn^b \quad \text{(since } c = 1)
\end{align*}
\]

2. Comparing logarithms. One slight oddness about the definition of Big-Oh is that it treats all logarithmic functions “the same.” Your task in this question is to investigate this, by proving the following statement:

\[
\forall a, b \in \mathbb{R}^+, a > 1 \land b > 1 \Rightarrow \log_a n \in O(\log_b n)
\]

We won’t ask you to expand the definition of Big-Oh, but if you aren’t quite sure, then we highly recommend doing so before attempting even your rough work!

Hint: look up the “change of base rule” for logarithms, if you don’t quite remember it!

Solution

Proof. Let \( a, b \in \mathbb{R}^+ \). Assume that \( a > 1 \) and \( b > 1 \). Let \( n_0 = 1 \), and let \( c = \frac{1}{\log_a b} \). [Since \( a, b > 1 \), we know that \( c > 0 \).] Let \( n \in \mathbb{N} \), and assume that \( n \geq n_0 \). We want to show that \( \log_a n \leq c \cdot \log_b n \).
The *change of base rule* tells us the following:

\[ \forall a, b, x \in \mathbb{R}^+, \; a \neq 1 \land b \neq 1 \Rightarrow \log_a x = \frac{\log_b x}{\log_b a} \]

(Note that when the bases are equal to 1, \( \log_a x \) is undefined when \( x \neq 1 \).)

Using this rule, we can write:

\[ \log_a n = \frac{\log_b n}{\log_b a} = \frac{1}{\log_b a} \log_b n = c \cdot \log_b n \]

Since we’ve proved that \( \log_a n = c \cdot \log_b n \), we can conclude that \( \log_a n \leq c \cdot \log_b n \).

[Note: we didn’t use the assumption that \( n \geq 1 \) in this proof.]
3. **Sum of functions.** Now let’s look at one of the most important properties of Big-Oh: how it behaves when adding functions together. Let \( f, g : \mathbb{N} \to \mathbb{R}^{\geq 0} \) (i.e., \( f \) and \( g \) are two functions that take natural numbers and return non-negative real numbers). We can define the **sum of \( f \) and \( g \)** as the function \( f + g : \mathbb{N} \to \mathbb{R}^{\geq 0} \) such that

\[
\forall n \in \mathbb{N}, \ (f + g)(n) = f(n) + g(n).
\]

For example, if \( f(n) = 2n \) and \( g(n) = n^2 + 3 \), then \( (f + g)(n) = 2n + n^2 + 3 \).

Consider the following statement:

\[
\forall f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}, \ g \in \mathcal{O}(f) \Rightarrow f + g \in \mathcal{O}(f)
\]

In other words, if \( g \) is Big-Oh of \( f \), then \( f + g \) is no bigger than just \( f \) itself, asymptotically speaking.

Your task for this question is to prove this statement. Keep in mind this is an implication: you’re going to assume that \( g \in \mathcal{O}(f) \), and you want to prove that \( f + g \in \mathcal{O}(f) \). It will likely be helpful to write out the full statement (with the definition of Big-Oh expanded), and use subscripts to help keep track of the variables.

**Solution**

Here’s the full statement, with the definitions expanded:

\[
\forall f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}, \ \left( \exists c, n_0 \in \mathbb{R}^+, \ \forall n \in \mathbb{N}, \ n \geq n_0 \Rightarrow g(n) \leq cf(n) \right) \Rightarrow \\
\left( \exists c_1, n_1 \in \mathbb{R}^+, \ \forall n \in \mathbb{N}, \ n \geq n_1 \Rightarrow f(n) + g(n) \leq c_1 f(n) \right)
\]

**Proof.** Let \( f, g : \mathbb{N} \to \mathbb{R}^{\geq 0} \). Assume that \( g \in \mathcal{O}(f) \), i.e., there exist \( n_0, c \in \mathbb{R}^+ \) such that for all natural numbers \( n \), if \( n \geq n_0 \) then \( g(n) \leq cf(n) \). We want to prove that \( f + g \in \mathcal{O}(f) \).

Let \( n_1 = n_0 \), and \( c_1 = c + 1 \). Let \( n \in \mathbb{N} \), and assume that \( n \geq n_1 \). We want to prove that \( f(n) + g(n) \leq c_1 f(n) \).

Since \( n \geq n_1 = n_0 \), by our assumption we know that \( g(n) \leq cf(n) \). So then:

\[
\begin{align*}
g(n) &\leq cf(n) \\
f(n) + g(n) &\leq f(n) + cf(n) \\
f(n) + g(n) &\leq (c + 1)f(n) \\
f(n) + g(n) &\leq c_1 f(n)
\end{align*}
\]