Solution Set

CIS 121—Data Structures and Algorithms—Fall 2020

Asymptotic Notation—Monday, September 14 / Tuesday, September 15

Readings

- Lecture Notes Chapter 5: Running Time and Growth Functions

Problems

Problem 0 [True or False]

1. A Big-O and Big-Omega bound for an algorithm correspond to worst-case and best-case runtime, respectively.
2. For any two functions, f and g, either $f \in O(g)$ or $g \in O(f)$.
3. $f(n) \in O(g(n))$ if and only if $g(n) \in \Omega(f(n))$.

Solution.

1. **False**. Big-O notation is a way to describe the limiting behavior of a function. One can provide a Big-O bound on the best, worst, and average case runtimes of an algorithm. It is not inherently tied to a particular one of these different ways to analyze an algorithm's efficiency. The same is true for Big-Omega.
2. **False**. Consider $\sin(x)$ and $\cos(x)$.
3. **True**. If $f(n) \in O(g(n))$, then we know there exist positive constants c and n_0 s.t. for all $n \geq n_0$

$$f(n) \leq c \cdot g(n)$$

Thus, for $c' = c^{-1}$ and $n_0' = n_0$ (both of which are positive), we have for all $n \geq n_0'$

$$g(n) \geq c' \cdot f(n)$$

Note: The other direction can be proven in an identical manner.

Problem 1

Prove that $3n^2 + 100n = \Theta(5n^2)$

Solution

We first prove Big-O. Recall the definition of Big-O—we wish to show that there exist positive constants c and n_0 such that for all $n \geq n_0$,

$$\begin{aligned} 3n^2 + 100n &\leq c \cdot 5n^2 \\ 3n + 100 &\leq 5c \cdot n \\ 100 &\leq (5c - 3)n \\ 100 &\leq (5(1) - 3)n && \text{(setting } c = 1\text{)} \\ 100 &\leq 2n \\ 50 &\leq n \end{aligned}$$

Since the expression holds for $c = 1$ and $n_0 = 50$, we have proved that $3n^2 + 100n = O(5n^2)$.
Next, we prove Big-Omega. Recall the definition of Big-Omega—we wish to show that there exist positive constants c and n_0 such that for all $n \geq n_0$,

$$\begin{aligned}
3n^2 + 100n &\geq c \cdot 5n^2 \\
3n^2 + 100n &\geq (3/5) \cdot 5n^2 && \text{(setting } c = 3/5\text{)} \\
3n^2 + 100n &\geq 3n^2 \\
100n &\geq 0 \\
n &\geq 0
\end{aligned}$$

Since the expression holds for $n_0 = 1$ and $c = 3/5$, we have proved that $3n^2 + 100n = \Omega(5n^2)$.

Since $3n^2 + 100n = O(5n^2)$ and $3n^2 + 100n = \Omega(5n^2)$, we have proved that $3n^2 + 100n = \Theta(5n^2)$.

Problem 2

Prove using induction that $n \log n = \Omega(n)$

Solution.

We will prove that $n \log n \geq c \cdot n, \ \forall n \geq n_0$ by using induction for $n_0 = 4$ and $c = 1$.

Base Case: $n = 4$. $4 \log 4 = 8 \geq 4$, so this holds.

Induction Hypothesis: Assume that $k \log k \geq k$ for some integer $k \geq 4$.

Induction Step: We need to show that $(k+1)\log(k+1) \geq k + 1$.

$$\begin{aligned}
(k+1)\log(k+1) &\geq (k+1)\log k && \text{(since } \log x \text{ is monotonically increasing)} \\
&= k \log k + \log k \\
&> k \log k + 1 && \text{(since } \log k \geq 2\text{)} \\
&\geq k + 1 && \text{(by IH)}
\end{aligned}$$

Problem 3

Prove that $\lg(n!) = \Theta(n \lg n)$.

Solution.

We first show $\lg(n!) = O(n \lg n)$:

Picking $c = 1$ and $n_0 = 1$, we have

$$\lg(n!) = \sum_{i=1}^{n} \lg i \leq n \lg n$$

This is clearly true for all $n \geq n_0$.

We then show that $\lg(n!) = \Omega(n \lg n)$:

To start, we find a simple lower-bound for $\lg n!$ that we can again lower-bound with some $c \cdot n \lg n$.

$$\begin{aligned}
\lg n! &= \lg 1 + \lg 2 + \cdots + \lg n \\
&\geq \lg \frac{n}{2} + \lg(\frac{n}{2} + 1) + \cdots + \lg n && \text{(the second half of the terms)} \\
&\geq \frac{n}{2} \cdot \lg \frac{n}{2} && \text{(since } \lg x \text{ is monotonically increasing)}
\end{aligned}$$

Choosing $c = \frac{1}{4}$ and $n_0 = 4$, it is clear that $\frac{n}{2} \lg \frac{n}{2} \geq \frac{n}{4} \lg n$ for all $n \geq n_0$ with some algebraic manipulation:

$$\begin{aligned} \frac{n}{2} \lg \frac{n}{2} &\geq \frac{n}{4} \lg n \\ \frac{n}{2} \lg n - \frac{n}{2} &\geq \frac{n}{4} \lg n \\ n \lg n &\geq 2n \\ \lg n &\geq 2 \end{aligned}$$

Therefore, $\lg(n!)$ is also $\Omega(n \lg n)$. $\square$