Asymptotic Notation and Mathematical Induction

K. Subramani
Department of Computer Science and Electrical Engineering,
West Virginia University,
Morgantown, WV
ksmani@csee.wvu.edu

1 Asymptotic Notation

The following list of notations is used to simplify the specification of algorithm complexity.

Definition: 1.1 $\mathbf{O}(\mathbf{g}(\mathbf{n})) = \{f(n) | \exists c, n_0 > 0,$ *such that* $f(n) \leq c.g(n) \forall n \geq n_0\}$ *i.e.* $g(n)$ *sits above* $f(n)$ *when their graphs are drawn, after some point* n_0. *Refer Figure (1).*

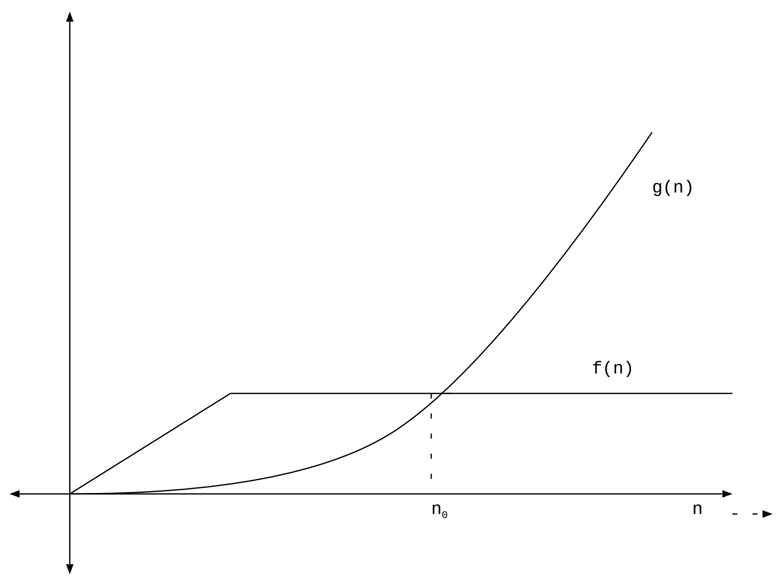

Figure 1: Growth of functions

Definition: 1.2 $\mathbf{\Omega}(\mathbf{g}(\mathbf{n})) = \{f(n) | \exists c, n_0 > 0,$ *such that* $f(n) \geq c.g(n) \forall n \geq n_0\}$ *i.e.* $g(n)$ *sits below* $f(n)$ *when their graphs are drawn, after some point* n_0.

Definition: 1.3 $\mathbf{\Theta}(\mathbf{g}(\mathbf{n})) = \{f(n) | \exists c_1, c_2, n_0 > 0,$ *such that* $c_1.g(n) \leq f(n) \leq c_2.g(n) \forall n \geq n_0\}$ *i.e.* $f(n)$ *is sandwiched by* $g(n)$ *for appropriately chosen constants* c_1 *and* c_2, *when their graphs are drawn, after some point* n_0.

Definition: 1.4 $\mathbf{o}(\mathbf{g}(\mathbf{n})) = \{f(n) | \lim_{n \to \infty} \frac{f(n)}{g(n)} = 0\}$ *i.e.* $g(n)$ *grows asymptotically faster than* $f(n)$

Definition: 1.5 $\omega(\mathbf{g}(\mathbf{n})) = \{f(n) | \lim_{n \to \infty} \frac{g(n)}{f(n)} = 0\}$ *i.e.* $f(n)$ *grows asymptotically faster than* $g(n)$

Remark: 1.1 *The asymptotic notation denotes a relationship and not a function i.e. the correct style is* $f(n) \in O(g(n))$ *and not* $f(n) = O(g(n))$*; however, over the years Computer Scientists have preferred the later notation scheme. But never forget that although we use equality it is actually a set inclusion symbol.*

Lemma: 1.1 $f(n) = O(g(n))$ *if and only if* $g(n) = \Omega(f(n))$

Lemma: 1.2 $f(n) = o(g(n))$ *if and only if* $g(n) = \omega(f(n))$

Lemma: 1.3 $f(n) = \Theta(g(n))$ *if and only if* $f(n) = O(g(n))$ *and* $f(n) = \Omega(g(n))$*.*

1.1 How to compare two polynomial functions

First, we can always write a polynomial in the form

$$a_d.n^d + a_{d-1}.n^{d-1} + \ldots a_0$$

as $\Theta(n^d)$. Note that a constant function is denoted by $O(1)$. Further, $O(1) \in O(n) \in O(n^2) \in O(n^3) \ldots \ldots ;$ Likewise $\ldots \in \Omega(n^d) \in \Omega(n^{d-1}) \in \ldots \Omega(1)$.

2 Laws of exponents

For all real $a \neq 0, m$ and n, we have

$$a^0 = 1. \tag{1}$$

$$a^1 = a. \tag{2}$$

$$a^{-1} = 1/a. \tag{3}$$

$$(a^m)^n = a^{m.n}. \tag{4}$$

$$(a^m)^n = (a^n)^m. \tag{5}$$

$$a^m.a^n = a^{m+n}. \tag{6}$$

Remark: 2.1 *Exponential functions grow at a much faster rate than any polynomial. Observe that, given* $a > 1$ *and* b*,*

$$\lim_{n \to \infty} \frac{n^b}{a^n} = 0, \tag{7}$$

by L'Hospital's rule. Hence, $n^b = o(a^n)$*.*

3 Laws of Logarithms

For all $a > 0, b > 0, c > 0$ and n

$$a = b^{\log_b a} \tag{8}$$

$$\log(ab) = \log a + \log b \tag{9}$$

$$\log a^n = n. \log a \tag{10}$$

$$\log_b a = \frac{1}{\log_a b} = \frac{\log_c a}{\log_c b} \tag{11}$$

$$a^{\log_b n} = n^{\log_b a} \tag{12}$$

3.1 Abbreviations

- $lg\ n = \log_2 n$
- $ln\ n = \log_e n$
- $\log^k n = (\log n)^k$
- $\log\ \log n = log(logn)$
- Iterated log:

$$\begin{aligned} \log^i n &= n, \ \ if \ \ i = 0, \\ &= \log(\log^{(i-1)} n), \ if \ \ i > 0 \ \ and \ \ \log^{(i-1)} n > 0. \end{aligned} \tag{13}$$

(14)

Then, we define

$$\log^* n = \min\{i \geq 0 : \log^{(i)} n \leq 1\}, \tag{15}$$

as the iterated log function. In other words, $\log^* n$ is the smallest integer i, such that after applying the log function i times repeatedly, the resultant quantity sinks below 1. The iterated log function has the smallest growth rate other than a constant function (remember that the constant function $f(n) = c$ does not grow at all!). When $n = 2^{65536}$, $\log^* n = 5$!!

Remark: 3.1 *Substitute* $\lg n$ *for* n *and* 2^a *for* a *in Equation (7), we get*

$$\lim_{n\to\infty} \frac{\lg^b n}{2^{a.\lg n}} = \lim_{n\to\infty} \frac{\lg^b n}{n^a} = 0 \tag{16}$$

Thus, we have

$$\lg^b n = o(n^a),$$

for any $a > 0$ *i.e. the log function does not grow as fast as polynomial functions. (Make sure that you can derive this equality).*

You now have the machinery to compare function growth rates.

4 Mathematical Induction

Mathematical induction is a general purpose tool of testing out a hyopothesis. The drawback of this technique is that the hypothesis must be supplied!

For example, let us say that we want a closed form representation of the sum

$$S_n = 1 + 2 + \ldots n = \sum_{i=1}^{n} i \tag{17}$$

Further assume that some good samaritan gave us the hypothesis that the closed form function is $\frac{n.(n+1)}{2}$. We need to check that the samaritan is not deceiving us.

So our Proposition is :

$$P(n) : S_n = \frac{n.(n+1)}{2}, \forall n = 1, 2, ...$$

Induction proceeds by two steps:

1. Step 1: The base case. Is $P(1)$ true? The Left Hand Side (LHS) is $S_1 = 1$ and the RHS is $\frac{1.(1+1)}{2}$, which is 1. Thus, LHS = RHS and $P(1)$ is true. Thus, the samaritan is correct at least when $n = 1$. This step is called *base confirmation.*

2. Step 2: The Inductive step: Here we assume that the samaritan is correct when $n = k$ i.e $P(k)$ is true and show that the formula must also hold for $n = k + 1$ i.e. $P(k + 1)$. Thus, we want to show that

$$P(k) \Rightarrow P(k+1)$$

In our case, assuming that $P(k)$ is true gives $S_k = \frac{k.(k+1)}{2}$. What is $P(k + 1)$? On the LHS it is

$$S_{k+1} = 1 + 2 + \dots k + (k+1)$$

$$\Rightarrow S_{k+1} = (1 + 2 + \dots k) + (k+1)$$

$$\Rightarrow S_{k+1} = S_k + (k+1)$$

Since $P(k)$ is true by assumption, we can set $S_k = \frac{k.(k+1)}{2}$

$$\Rightarrow S_{k+1} = \frac{k.(k+1)}{2} + (k+1)$$

$$\Rightarrow S_{k+1} = \frac{(k+1).(k+2)}{2}$$

which is exactly what we would obtain by substituting $(k + 1)$ in the samaritan's formula (= RHS). Thus we have,

$$P(k) \Rightarrow P(k+1).$$

Since $P(1)$ is true, we can then conclude that $P(2)$ is true and hence $P(3)$ is true and .... $P(n)$ is true for all n. Thus the samaritan was good after all.

Remark: 4.1 *Show that*

$$\sum_{i=1}^{n} i^2 = \frac{n.(n+1).(2.n+1)}{6}$$