# Chapter 5

# Analyzing Algorithms

## 5.1 Run time and constant factors

When calculating the running time of a program, we may know how many basic "steps" it takes as a function of input size, but we may not know how long each step takes on a particular computer. We would like to estimate the overall running time of an algorithm while ignoring constant factors (like how fast the CPU is). So, for example, if we have 3 machines, where operations take $3\mu s$, $8\mu s$ and $0.5\mu s$, the three functions measuring the amount of time required, $t(n) = 3n^2$, $t(n) = 8n^2$, and $t(n) = n^2/2$ are considered the same, ignoring ("to within") constant factors (the time required always grows according to a quadratic function in terms of the size of the input $n$).

The nice thing is that this means that lower order terms can be ignored as well! So $f(n) = 3n^2$ and $g(n) = 3n^2 + 2$ are considered "the same", as are $h(n) = 3n^2 + 2n$ and $j(n) = 5n^2$. Notice that

$$\forall n \in \mathbb{N}, n \geq 1 \Rightarrow f(n) \leq g(n) \leq h(n) \leq j(n)$$

but there's always a constant factor that can reverse any of these inequalities.

Really what we want to measure is the growth rate of functions (and in computer science, the growth rate of functions that bound the running time of algorithms). You might be familiar with binary search and linear search (two algorithms for searching for a value in a sorted array). Suppose one computer runs binary search and one computer runs linear search. Which computer will give an answer first, assuming the two computers run at roughly the same CPU speed? What if one computer is much faster (in terms of CPU speed) than the other, does it affect your answer? What if the array is really, really big?

## How large is "sufficiently large?"

Is binary search a better algorithm than linear search?[1] It depends on the size of the input. For example, suppose you established that linear search has complexity $L(n) = 3n$ and binary search has complexity $B(n) = 9\log_2 n$. For the first few $n$, $L(n)$ is smaller than $B(n)$. However, certainly for $n > 10$, $B(n)$ is smaller, indicating less "work" for binary search.

When we say "large enough" $n$, we mean we are discussing the asymptotic behaviour of the complexity function, and we are prepared to ignore the behaviour near the origin.

## 5.2 Asymptotic notation: Making Big-O precise

We define $\mathbb{R}^{\geq 0}$ as the set of nonnegative real numbers, and define $\mathbb{R}^{+}$ as the set of positive real numbers. Now here's the precise definition of "The set of functions that, ignoring a constant, are eventually no more than $f$":

Definition: For any function $f : \mathbb{N} \to \mathbb{R}^{\geq 0}$ (i.e., any function mapping naturals to nonnegative reals), let

$$O(f) = \{g : \mathbb{N} \to \mathbb{R}^{\geq 0} \mid \exists c \in \mathbb{R}^{+}, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow g(n) \leq cf(n)\}.$$

Saying $g \in O(f)$ says that "$g$ grows no faster than $f$" (or equivalently, "$f$ is an upper bound for $g$), so long as we modify our understanding of "growing no faster" and being an "upper bound" with the practice of ignoring constant factors. Now we can prove some theorems.

Suppose $g(n) = 3n^2 + 2$ and $f(n) = n^2$. Then $g \in O(f)$. We need to prove that $\exists c \in \mathbb{R}^{+}, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow 3n^2 + 2 \leq cn^2$. It's enough to find some $c$ and $B$ that "work" in order to prove the theorem.

Finding $c$ means finding a factor that will scale $n^2$ up to the size of $3n^2 + 2$. Setting $c = 3$ almost works, but there's that annoying additional term 2. Certainly $3n^2 + 2 < 4n^2$ so long as $n \geq 2$, since $n \geq 2 \Rightarrow n^2 > 2$. So pick $c = 4$ and $B = 2$ (other values also work, but we like the ones we thought of first). Now concoct a proof of

$$\exists c \in \mathbb{R}^{+}, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow 3n^2 + 2 \leq cn^2.$$

Let $c = 4$.
Then $c \in \mathbb{R}^{+}$.
Let $B = 2$.
Then $B \in \mathbb{N}$.
Assume $n \in \mathbb{N}$. (we assume that $n$ is an arbitrary natural number)
  Assume $n \geq B$.
    Then $n^2 \geq B^2 = 4$. (squaring is monotonic on natural numbers)
    So $n^2 \geq 2$.
    So $3n^2 + n^2 \geq 3n^2 + 2$. (adding $3n^2$ to both sides of the inequality)
    So $3n^2 + 2 \leq 4n^2$.
  Thus, $n \geq B \Rightarrow 3n^2 + 2 \leq 4n^2$. (by $\Rightarrow$I)
Since $n$ is an arbitrary natural number, $\forall n \in \mathbb{N}, n \geq B \Rightarrow 3n^2 + 2 \leq 4n^2$. (by $\forall$I)
Since $B$ is a natural number, $\exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow 3n^2 + 2 \leq cn^2$. (by $\exists$I)
Since $c$ is a positive real number, $\exists c \in \mathbb{R}^{+}, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow 3n^2 + 2 \leq cn^2$. (by $\exists$I)

So, by definition, $g \in O(f)$.

Now suppose that $g(n) = n^4$ and $f(n) = 3n^2$. Is $g \in O(f)$? No. We can see intuitively that any constant that we multiply times $3n^2$ will be overwhelmed by the extra factor of $n^2$ in $g(n)$. But to show this clearly, we negate the definition and then prove the negation:

$$\forall c \in \mathbb{R}^{+}, \forall B \in \mathbb{N}, \exists n \in \mathbb{N}, n \geq B \wedge n^4 > c3n^2.$$

The parameter we have some control over is $n$, and we need to pick it so that $n \geq B$ and $n^4 > c3n^2$. Solve for $n$:

$$\begin{aligned} & n^4 > c3n^2 \\ \Leftrightarrow \quad & n^4/n^2 > c3n^2/n^2 \quad \text{(when } n > 0\text{)} \\ \Leftrightarrow \quad & n^2 > 3c \\ \Leftrightarrow \quad & n > \sqrt{3c}. \end{aligned}$$

Notice that we were reasoning backwards (bottom up) here: we need to pick a condition on $n$ so that $n^4 > c3n^2$ would hold (hence why everything had to be an equivalence). Notice also that we needed to assume that $n > 0$ (to avoid division by zero). So to satisfy the conditions, we can set $n = B + \lceil \sqrt{3c} \rceil + 1$. Since $\sqrt{3c}$ is not necessarily a natural number, we take its ceiling. Now we can generate the proof.

> Assume $c \in \mathbb{R}^+$.
>> Assume $B \in \mathbb{N}$.
>>> Let $n = B + \lceil \sqrt{3c} \rceil + 1$.
>>> Then $n \in \mathbb{N}$. (since $B \in \mathbb{N}$, $1 \in \mathbb{N}$, and $\lceil \sqrt{3c} \rceil \in \mathbb{N}$ (since $c > 0$) and $\mathbb{N}$ is closed under sums).
>>> So $n \geq B$ (since it is the sum of $B$ and two other non-negative numbers).
>>> So $n \geq \lceil \sqrt{3c} \rceil + 1$. (since $B \geq 0$)
>>> So $n^2 > (\lceil \sqrt{3c} \rceil)^2$. (dropping a positive term, squaring both sides)
>>> So $n^2 > 3c$. (dropping the ceiling)
>>> So $n^4 > 3cn^2$. (multiplying both sides by $n^2$)
>>> Since $n$ is a natural number, $\exists n \in \mathbb{N}, n \geq B \wedge n^4 > c3n^2$. (by $\exists$I)
>>
>> Since $B$ is an arbitrary element of $\mathbb{N}$, $\forall c \in \mathbb{R}^+, \forall B \in \mathbb{N}, \exists n \in \mathbb{N}, n \geq B \wedge n^4 > c3n^2$. (by $\exists$I)
>
> Since $c$ is an arbitrary element of $\mathbb{R}^+$, $\forall c \in \mathbb{R}^+, \forall B \in \mathbb{N}, \exists n \in \mathbb{N}, n \geq B \wedge n^4 > c3n^2$. (by $\exists$I)
> By definition, this means that $g \notin O(f)$.

## A MORE COMPLEX EXAMPLE

Let's prove that $2n^3 - 5n^4 + 7n^6$ is in $O(n^2 - 4n^5 + 6n^8)$. We begin with:

> Let $c =$ ____. Then $c \in \mathbb{R}^+$.
> Let $B =$ ____. Then $B \in \mathbb{N}$.
> Assume $n \in \mathbb{N}$.
>> Assume $n \geq B$.
>>> Then $2n^3 - 5n^4 + 7n^6 \leq \cdots \leq c(n^2 - 4n^5 + 6n^8)$.
>>
>> Thus $n \geq B \Rightarrow 2n^3 - 5n^4 + 7n^6 \leq c(n^2 - 4n^5 + 6n^8)$.
>
> Since $n$ is an arbitrary natural number, $\forall n \in \mathbb{N}, n \geq B \Rightarrow 2n^3 - 5n^4 + 7n^6 \leq c(n^2 - 4n^5 + 6n^8)$.
> Since $B$ is a natural number, and since $c$ is a positive real number,
> $\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow 2n^3 - 5n^4 + 7n^6 \leq c(n^2 - 4n^5 + 6n^8)$.

To fill in the $\cdots$ we try to form a chain of inequalities, working from both ends, simplifying the expressions:

$$
\begin{aligned}
2n^3 - 5n^4 + 7n^6 &\leq 2n^3 + 7n^6 \quad \text{(drop $-5n^4$ because it doesn't help us in an important way)} \\
&\leq 2n^6 + 7n^6 \quad \text{(increase $n^3$ to $n^6$ because we have to handle $n^6$ anyway)} \\
&= 9n^6 \\
&\leq 9n^8 \qquad \text{(simpler to compare)} \\
&= 2(9/2)n^8 \quad \text{(get as close to form of the simplified end result: now choose $c = 9/2$)} \\
&= 2cn^8 \\
&= c(-4n^8 + 6n^8) \quad \text{(reading bottom up: decrease $-4n^5$ to $-4n^8$ because we have to handle $n^8$ anyway)} \\
&\leq c(-4n^5 + 6n^8) \quad \text{(reading bottom up: drop $n^2$ because it doesn't help us in an important way)} \\
&\leq c(n^2 - 4n^5 + 6n^8)
\end{aligned}
$$

We never needed to restrict $n$ in any way beyond $n \in \mathbb{N}$ (which includes $n \geq 0$), so now fill in $c = 9/2$, $b = 0$, and complete the proof.

Let's use this approach to reprove $n^4 \notin O(3n^2)$.

Assume $c \in \mathbb{R}^+$.
  Assume $B \in \mathbb{N}$.
    Let $n =$ ____.
    ...
    So $n \in \mathbb{N}$.
    ...
    So $n \geq B$.
    ...
    So $n^4 > c3n^2$.
Thus $\forall c \in \mathbb{R}^+, \forall B \in \mathbb{N}, \exists n \in \mathbb{N}, n \geq B \wedge n^4 > c3n^2$.

Here's our chain of inequalities (the third $\cdots$):

$$\begin{aligned} \text{And } n^4 &\geq n^3 && \text{(don't need full power of } n^4\text{)} \\ &= n \cdot n^2 && \text{(make form as close as possible)} \\ &> c \cdot 3n^2 && \text{(if we make } n > 3c \text{ and } n > 0\text{).} \end{aligned}$$

Now pick $n = \max(B, \lceil 3c + 1 \rceil)$.
The first $\cdots$ is:

Since $c > 0$, $3c + 1 > 0$, so $\lceil 3c + 1 \rceil \in \mathbb{N}$.
Since $B \in \mathbb{N}$, $\max(B, \lceil 3c + 1 \rceil) \in \mathbb{N}$.

The second $\cdots$ is:

$\max(B, \lceil 3c + 1 \rceil) \geq B$.

We also note just before the chain of inequalities:

$n = \max(B, \lceil 3c + 1 \rceil) \geq \lceil 3c + 1 \rceil \geq 3c + 1 > 3c$.

Some points to note are:

- Don't "solve" for $n$ until you've made the form of the two sides as close as possible.
- You're not exactly solving for $n$: you are finding a condition of the form $n >$ __ that makes the desired inequality true. You might find yourself using the max function a lot.
- Be careful that you aren't "solving" for $n$ in the wrong direction: the first time we reasoned that $n^4 > c3n^2 \Rightarrow n > \sqrt{3c}$, but the proof needs the reverse direction. Luckily, each of the steps were reversible (i.e., they were all equivalences), yielding the needed line of reasoning.

## 5.3 Other asymptotic notations: $\Omega$ and $\Theta$

In analogy with $O(f)$, consider two other definitions:

Definition: For any function $f : \mathbb{N} \to \mathbb{R}^{\geq 0}$, let

$$\Omega(f) = \{g : \mathbb{N} \to \mathbb{R}^{\geq 0} \mid \exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow g(n) \geq cf(n)\}.$$

To say "$g \in \Omega(f)$" expresses the concept that "$g$ grows at least as fast as $f$." ($f$ is a lower bound on $g$).

DEFINITION: For any function $f : \mathbb{N} \to \mathbb{R}^{\geq 0}$, let

$$\Theta(f) = \{g : \mathbb{N} \to \mathbb{R}^{\geq 0} \mid \exists c_1 \in \mathbb{R}^+, \exists c_2 \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow c_1 f(n) \leq g(n) \leq c_2 f(n)\}.$$

To say "$g \in \Theta(f)$" expresses the concept that "$g$ grows at the same rate as $f$." ($f$ is a tight bound for $g$, or $f$ is both an upper bound and a lower bound on $g$).

## 5.4 Mathematical induction

Suppose $P(n)$ is some predicate of the natural numbers, and:

$$(*) \quad P(0) \wedge (\forall n \in \mathbb{N}, P(n) \Rightarrow P(n+1)).$$

You should certainly be able to show that (*) implies $P(0)$, $P(1)$, $P(2)$, in fact $P(n)$ where $n$ is any natural number you have the patience to follow the chain of results to obtain. In fact, we feel that we can "turn the crank" enough times to show that (*) implies $P(n)$ for any natural number $n$. In other words,

$$[P(0) \wedge (\forall n \in \mathbb{N}, P(n) \Rightarrow P(n+1))] \Rightarrow [\forall n \in \mathbb{N}, P(n)].$$

This is called the Principle of Simple Induction (PSI). It isn't proved, it is an axiom that we assume to be true.

Here's an an application of the PSI that will be useful for some big-Oh problems.

$P(n)$: $2^n \geq 2n$.

I'd like to prove that $\forall n, P(n)$, using the PSI. Here's what I do:

PROVE $P(0)$: $P(0)$ states that $2^0 = 1 \geq 2(0) = 0$, which is true.

PROVE $\forall n \in \mathbb{N}, P(n) \Rightarrow P(n+1)$:

Assume $n \in \mathbb{N}$. ($n$ is an arbitrary natural number)
  Assume $P(n)$, that is $2^n \geq 2n$. (the antecedent)
    Then $n = 0 \vee n > 0$. (natural numbers are non-negative)
    CASE 1: Assume $n = 0$.
      Then $2^{n+1} = 2^1 = 2 \geq 2(n+1) = 2$.
    CASE 2: Assume $n > 0$.
      Then $n \geq 1$. (since $n$ is an integer greater than 0)
      Then $2^n \geq 2$. (since $n \geq 1$, and $2^n$ is monotone increasing)
      Then $2^{n+1} = 2^n + 2^n \geq 2n + 2 = 2(n+1)$. (by previous line and IH $P(n)$)
    Then $2^{n+1} \geq 2(n+1)$, which is $P(n+1)$. (since it is true in both possible cases)
  Then $P(n) \Rightarrow P(n+1)$. (by $\Rightarrow$I)
Then $\forall n \in \mathbb{N}, P(n) \Rightarrow P(n+1)$. (by $\forall$I)

I now conclude, by the PSI, $\forall n \in \mathbb{N}, P(n)$, that is $2^n \geq 2n$.

Here's a big-Oh problem where I can use $P(n)$. Let $g(n) = 2^n$ and $f(n) = n$. I want to show that $g \notin O(f)$.

Assume $c \in \mathbb{R}^+$.

Assume $B \in \mathbb{N}$.

Let $k = \lceil \log_2(c) \rceil + 1 + B$. Let $n = 2k$.

Then $n \in \mathbb{N}$. (since $\lceil c \rceil, 1, 2, B \in \mathbb{N}$, $\mathbb{N}$ closed under $+, \times$)

Then $n \geq B$. ($n$ is at least twice $B$)

Then $2^k > c$. (by choice of $k$, and since $2^x$ is increasing function)

Then

$$\begin{aligned} g(n) = 2^n &= 2^k \times 2^k && \text{(by choice of } k\text{)} \\ &\geq 2^k \times 2k && \text{(by } P(2k)\text{)} \\ &= 2^k \times n > cn && \text{(by } n = 2k \text{ and } 2^k > c\text{)} \\ &= cf(n) \end{aligned}$$

Then $n \geq B \wedge g(n) \geq cf(n)$. (by $\wedge$I)

Then $\exists n \in \mathbb{N}, n \geq B \wedge g(n) \geq cf(n)$. (by $\exists$I)

Thus $\forall c \in \mathbb{R}, \forall B \in \mathbb{N}, \exists n \in \mathbb{N}, n \geq B \wedge g(n) > cf(n)$. (by $\forall$I)

So, I can conclude that $g \notin O(f)$.

What happens to induction for predicates that are true for all natural numbers after a certain point, but untrue for the first few natural numbers? For example, $2^n$ grows much more quickly than $n^2$, but $2^3$ is not larger than $3^2$. Choose $n$ big enough, though, and it is true that:

$$P(n) : 2^n > n^2.$$

You can't prove this for all $n$: it is false for $n = 2, n = 3$, and $n = 4$. So you'll need to restrict the domain and prove that for all natural numbers greater than 4, $P(n)$ is true. We don't have a slick way to restrict domains in our symbolic notation. Let's consider three ways to restrict the natural numbers to just those greater than 4, and then use induction.

Restrict using implication: Our first method of restriction uses implication to restrict the domain where we claim $P(n)$ is true:

$$\forall n \in \mathbb{N}, n \geq 5 \Rightarrow P(n).$$

The expanded predicate $Q(n) : n \geq 5 \Rightarrow P(n)$ now fits our pattern for simple induction, and all we need to do is prove:

1. $Q(0)$ is true (it is vacuously true, since $0 \geq 5$ is false).
2. $\forall n \in \mathbb{N}, Q(n) \Rightarrow Q(n+1)$. This breaks into cases. If $n < 4$, then $Q(n)$ and $Q(n+1)$ are both vacuously true (the antecedents of the implication are false, since $n$ and $n+1$ are not greater than, nor equal to, 5), so there is nothing to prove. If $n = 4$, then $Q(n)$ is vacuously true, but $Q(n+1)$ has a true antecedent ($5 \geq 5$), so we need to prove $Q(5)$ directly: $2^5 > 5^2$ is true, since $32 > 25$. For $n > 5$, we can depend on the assumption of the consequent of $Q(n-1)$ being true to prove $Q(n)$:

$$\begin{aligned} 2^n &= 2^{n-1} + 2^{n-1} && \text{(definition of } 2^n\text{)} \\ &> 2(n-1)^2 && \text{(antecedent of } Q(n-1)\text{)} \\ &= 2n^2 - 2n + 2 = n^2 + n(n-2) + 2 \geq n^2 + 2 > n^2 && \text{(since } n > 4 \geq 2\text{)} \end{aligned}$$

After all that work, it turns out that we need prove just two things:

1. $P(5)$
2. $\forall n \in \mathbb{N}$, if $n > 4$, then $P(n) \Rightarrow P(n+1)$.

This is the same as before, except now our base case is $P(5)$ rather than $P(0)$, and we get to use the fact that $n \geq 5$ in our induction step (if we need it).

RESTRICT BY TRANSLATION: We can also restrict the domain by translating our predicate, so that $Q(n) = P(n+5)$, that is:

$$Q(n) : 2^{n+5} > (n+5)^2$$

Now our task is to prove $Q(0)$ is true (it is: $32 > 25$), and that for all $n \in \mathbb{N}$, $Q(n) \Rightarrow Q(n+1)$. This is simple induction.

RESTRICT BY SET DIFFERENCE: A third way to restrict the domain is by set difference:

$$\forall n \in \mathbb{N} \setminus \{0, 1, 2, 3, 4\}, P(n)$$

Again, we'll need to prove $P(5)$, and then that $\forall n \in \mathbb{N} \setminus \{0, 1, 2, 3, 4\}, P(n) \Rightarrow P(n+1)$.

Whichever argument you're comfortable with, notice that simple induction is basically the same: you prove the base case (which may now be greater than 0), and you prove the induction step.

## 5.5 Some theorems about asymptotic notation

Here are some general results that we now have the tools to prove.

- $f \in O(f)$.
- $(f \in O(g) \wedge g \in O(h)) \Rightarrow f \in O(h)$.
- $g \in \Omega(f) \Leftrightarrow f \in O(g)$.
- $g \in \Theta(f) \Leftrightarrow g \in O(f) \wedge g \in \Omega(f)$.

Test your intuition about Big-O by doing the "scratch work" to answer the following questions:

- Are there functions $f, g$ such that $f \in O(g)$ and $g \in O(f)$ but $f \neq g$?[2]
- Are there functions $f, g$ such that $f \notin O(g)$, and $g \notin O(f)$?[3]

To show that $(f \in O(g) \wedge g \in O(h)) \Rightarrow f \in O(h)$, we need to find a constant $c \in \mathbb{R}^+$ and a constant $B \in \mathbb{N}$, that satisfy:

$$\forall n \in \mathbb{N}, n \geq B \Rightarrow f(n) \leq ch(n).$$

Since we have constants that scale $h$ to $g$ and then $g$ to $f$, it seems clear that we need their product to scale $g$ to $f$. And if we take the maximum of the two starting points, we can't go wrong. Making this precise:

THEOREM 1: For any functions $f, g, h : \mathbb{N} \rightarrow \mathbb{R}^{\geq 0}$, we have $(f \in O(g) \wedge g \in O(h)) \Rightarrow f \in O(h)$.

PROOF:

Assume $f \in O(g) \wedge g \in O(h)$.

So $f \in O(g)$.
So $g \in O(h)$.
So $\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n > B \Rightarrow f(n) \leq cg(n)$. (by defn. of $f \in O(g)$).
Consider $c_g \in \mathbb{R}^+, B_g \in \mathbb{N}$ such that $\forall n \in \mathbb{N}, n \geq B \Rightarrow f(n) \leq c_g g(n)$.
So $\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow g(n) \leq ch(n)$. (by defn. of $g \in O(h)$).
Consider $c_h \in \mathbb{R}^+, B_h \in \mathbb{N}$ such that $\forall n \in \mathbb{N}, n \geq B_h \Rightarrow g(n) \leq c_h h(n)$.
Let $c = c_g c_h$. Then $c \in \mathbb{R}^+$.
Let $B = \max(B_g, B_h)$. Then $B \in \mathbb{N}$.
Assume $n \in \mathbb{N}$.
Assume $n \geq B$.
Then $n \geq B_h$ (definition of max), so $g(n) \leq c_h h(n)$.
Then $n \geq B_g$ (definition of max), so $f(n) \leq c_g g(n) \leq c_g c_h h(n)$.
So $f(n) \leq ch(n)$.
So $n \geq B \Rightarrow f(n) \leq ch(n)$.
Since $n$ is an arbitrary natural number, $\forall n \in \mathbb{N}, n \geq B \Rightarrow f(n) \leq ch(n)$.
Since $c$ is a positive real number, and since $B$ is a natural number,
$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow f(n) \leq ch(n)$.
So $f \in O(g)$, by definition.
So $(f \in O(g) \wedge g \in O(h)) \Rightarrow f \in O(h)$.

To show that $g \in \Omega(f) \Leftrightarrow f \in O(g)$, it is enough to note the the constant, $c$, for one direction is positive, so its reciprocal will work for the other direction.[4]

Theorem 2: For any functions $f, g : \mathbb{N} \rightarrow \mathbb{R}^{\geq 0}$, we have $g \in \Omega(f) \Leftrightarrow f \in O(g)$.

Proof:
$g \in \Omega(f)$
$\Leftrightarrow$ (definition)
$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow g(n) \geq cf(n)$
$\Leftrightarrow$ (by letting $c' = 1/c$ and $B' = B$).
$\exists c' \in \mathbb{R}^+, \exists B' \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B' \Rightarrow f(n) \leq c' g(n)$
$\Leftrightarrow$ (definition)
$f \in O(g)$

To show $g \in \Theta(f) \Leftrightarrow g \in O(f) \wedge g \in \Omega(f)$, it's really just a matter of unwrapping the definitions.

Theorem 3: For any functions $f, g : \mathbb{N} \rightarrow \mathbb{R}^{\geq 0}$, we have $g \in \Theta(f) \Leftrightarrow g \in O(f) \wedge g \in \Omega(f)$.

Proof:
$g \in \Theta(f)$
$\Leftrightarrow$ (definition)
$\exists c_1 \in \mathbb{R}^+, \exists c_2 \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow c_1 f(n) \leq g(n) \leq c_2 f(n)$.
$\Leftrightarrow$ (combined inequality, and $B = \max(B_1, B_2)$).
$\exists c_1 \in \mathbb{R}^+, \exists B_1 \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B_1 \Rightarrow g(n) \geq c_1 f(n) \wedge \exists c_2 \in \mathbb{R}^+, \exists B_2 \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B_2 \Rightarrow g(n) \leq c_2 f(n)$
$\Leftrightarrow$ (definition)
$g \in \Omega(f) \wedge g \in O(f)$

### Taxonomy of results

A LEMMA is a small result needed to prove something we really care about. A THEOREM is the main result that we care about (at the moment). A COROLLARY is an easy (or said to be easy) consequence of another result. A CONJECTURE is something suspected to be true, but not yet proven.

Here's an example of a conjecture whose proof has evaded the best minds for over 70 years. Maybe you'll prove it.

Define $f(n)$, for $n \in \mathbb{N}$ by:

$$f(n) = \begin{cases} n/2, & n \text{ even} \\ 3n+1, & n \text{ odd} \end{cases}$$

Let's define $f^2(n)$ as $f(f(n))$, and define $f^{k+1}(n)$ as $f(f^k(n))$. Here's the conjecture:

CONJECTURE: $\forall n \in \mathbb{N}, \exists k \in \mathbb{N}, f^k(n) = 1$.

Easy to state, but (so far) hard to prove or disprove.

Here's an example of a corollary that recycles some of the theorems we've already proven (so we don't have to do the grubby work). To show $g \in \Theta(f) \Leftrightarrow f \in \Theta(g)$, I re-use theorems proved above and the commutativity of $\wedge$:

COROLLARY: For any functions $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$, we have $g \in \Theta(f) \Leftrightarrow f \in \Theta(g)$.

PROOF:

$g \in \Theta(f)$
$\Leftrightarrow$ (by Theorem 3)
$g \in O(f) \wedge g \in \Omega(f)$.
$\Leftrightarrow$ (by Theorem 2)
$g \in O(f) \wedge f \in O(g)$
$\Leftrightarrow$ (by commutativity of $\wedge$)
$f \in O(g) \wedge g \in O(f)$
$\Leftrightarrow$ (by Theorem 2)
$f \in O(g) \wedge f \in \Omega(g)$
$\Leftrightarrow$ (by Theorem 3)
$f \in \Theta(g)$.

## 5.6 A very important note about asymptotic notation

Note that asymptotic notation (the Big-O, Big-$\Omega$, and Big-$\Theta$ definitions) bound the asymptotic growth rates of *functions*, as $n$ approaches infinity. Often in computer science we use this asymptotic notation to bound functions that express the running times of algorithms, perhaps in best case or in worst case. Asymptotic notation *does not* express or bound the worst case or best case running time, only the functions expressing these values.

This distinction is subtle, but crucial to understanding both running times and asymptotic notation. If this warning doesn't seem important to you now, come back and read this again in a few weeks, months, or courses. You'll thank me later.

## Exercises

1. Prove or disprove the following claims:

   (a) $7n^3 + 11n^2 + n \in O(n^3)$ [5]

   (b) $n^2 + 165 \in \Omega(n^4)$

   (c) $n! \in O(n^n)$

   (d) $n \in O(n \log_2 n)$

   (e) $\forall k \in \mathbb{N}, k > 1 \Rightarrow \log_k n \in \Theta(\log_2 n)$

2. Define $g(n) = \begin{cases} n^3/165, & n < 165 \\ \left\lceil \sqrt{6n^5} \right\rceil, & n \geq 165 \end{cases}$. Note that $\forall x \in \mathbb{R}, x \leq \lceil x \rceil < x + 1$.

   Prove that $g \in O(n^{2.5})$.

3. Let $\mathcal{F}$ be the set of functions from $\mathbb{N}$ to $\mathbb{R}^{\geq 0}$. Prove the following theorems:

   (a) For $f, g \in \mathcal{F}$, if $g \in \Omega(f)$ then $g^2 \in \Omega(f^2)$.

   (b) $\forall k \in \mathbb{N}, k > 1 \Rightarrow \forall d \in \mathbb{R}^+, d \log_k n \in \Theta(\log_2 n)$. [6]

   Notice that (b) means that all logarithms eventually grow at the same rate (up to a multiplicative constant), so the base doesn't matter (and can be omitted inside the asymptotic notation).

4. Let $\mathcal{F}$ be the set of functions from $\mathbb{N}$ to $\mathbb{R}^{\geq 0}$. Prove or disprove the following claims:

   (a) $\forall f \in \mathcal{F}, \forall g \in \mathcal{F}, f \in O(g) \Rightarrow (f + g) \in \Theta(g)$

   (b) $\forall f \in \mathcal{F}, \forall f' \in \mathcal{F}, \forall g \in \mathcal{F}, (f \in O(g) \wedge f' \in O(g)) \Rightarrow (f + f') \in O(g)$

5. For each function $f$ in the left column, choose one expression $O(g)$ from the right column such that $f \in O(g)$. Use each expression exactly once.

   (i) $3 \cdot 2^n \in$ ________

   (ii) $\frac{2n^4+1}{n^3+2n-1} \in$ ________

   (iii) $(n^5 + 7)(n^5 - 7) \in$ ________

   (iv) $\frac{n^4 - n \log_2 n}{n^2+1} \in$ ________

   (v) $\frac{n \log_2 n}{n-5} \in$ ________

   (vi) $8 + \frac{1}{n^2} \in$ ________

   (vii) $2^{3n+1} \in$ ________

   (viii) $n! \in$ ________

   (ix) $\frac{5 \log_2(n+1)}{1 + n \log_2 3n} \in$ ________

   (x) $(n-2)\log_2(n^3+4) \in$ ________

   (a) $O(\frac{1}{n})$

   (b) $O(1)$

   (c) $O(\log_2 n)$

   (d) $O(n)$

   (e) $O(n \log_2 n)$

   (f) $O(n^2)$

   (g) $O(n^{10})$

   (h) $O(2^n)$

   (i) $O(10^n)$

   (j) $O(n^n)$

# Chapter 5 Notes

[1]Better in the sense of time complexity.

[2]Sure, $f = n^2$, $g = 3n^2 + 2$.

[3]Sure. $f$ and $g$ don't need to both be monotonic, so let $f(n) = n^2$ and

$$g(n) = \begin{cases} n, & n \text{ even} \\ n^3, & n \text{ odd} \end{cases}$$

So not every pair of functions from $\mathbb{N} \to \mathbb{R}^{\geq 0}$ can be compared using Big-O.

[4]Let's try the symmetrical presentation of bi-implication.

[5]The claim is true.
Let $c = 8$. Then $c \in \mathbb{R}^+$.
Let $B = 12$. Then $B \in \mathbb{N}$.
Assume $n \in \mathbb{N}$.
Assume $n \geq B$.
Then $n^3 = n \times n^2 \geq 12 \times n^2 = 11 \times n^2 + n^2$. (since $n \geq B = 12$).
So $n^2 \geq 12n$. (since $n \geq 12$, multiplying both sides by $n > 0$).
So $12 > 1 \Rightarrow 12n > n$. (Multiplying both sides by $n > 0$).
So $n^3 \geq 12n^2 = 11n^2 + n^2 \geq 11n^2 + 12n \geq 11n^2 + n$.
So $7n^3 \geq 7n^3$.
Thus $cn^3 = 8n^3 = 7n^3 + n^3 \geq 7n^3 + 11n^2 + n$. (adding the two inequalities).
So $n \geq B \Rightarrow 7n^3 + 11n^2 + n \leq cn^3$.
Since $n$ is an arbitrary element of $\mathbb{N}$, $\forall n \in \mathbb{N}$, $n \geq B \Rightarrow 7n^3 + 11n^2 + n \leq cn^3$.
Since $B$ is a natural number, $\exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow 7n^3 + 11n^2 + n \leq cn^3$.
Since $c$ is a real positive number, $\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow 7n^3 + 11n^2 + n \leq cn^3$.
By definition, $7n^3 + 11n^2 + n \in O(n^3)$.

[6]Assume $k \in \mathbb{N}$ and assume $k > 1$.
Assume $d \in \mathbb{R}^+$.
It suffices to argue that $d \log_k n \in \Theta(\log_2 n)$.
Let $c_1 = \frac{d}{\log_2 k}$. Since $k > 1$, $\log_2 k \neq 0$ and so $c_1 \in \mathbb{R}^+$.
Let $c_2 = \frac{d}{\log_2 k}$. Then $c_2 \in \mathbb{R}^+$.
Let $B = 1$. Then $B \in \mathbb{N}$.
Assume $n \in \mathbb{N}$.
Assume that $n \geq B$.
Then $c_1 \log_2 n = \frac{d}{\log_2 k} \log_2 n = d\frac{\log_2 n}{\log_2 k} = d \log_k n \leq d \log_k n$.
Moreover, $d \log_k n \leq d\frac{\log_2 n}{\log_2 k} = \frac{d}{\log_2 k} \log_2 n = c_2 \log_2 n$.
So $n \geq B \Rightarrow c_1 \log_2 n \leq d \log_k n \leq c_2 \log_2 n$.
Since $n$ is arbitrary, $\forall n \in \mathbb{N}, n \geq B \Rightarrow c_1 \log_2 n \leq d \log_k n \leq c_2 \log_2 n$.
Thus $\exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow c_1 \log_2 n \leq d \log_k n \leq c_2 \log_2 n$.
Thus $\exists c_1 \in \mathbb{R}^+, \exists c_2 \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow c_1 \log_2 n \leq d \log_k n \leq c_2 \log_2 n$.
By definition, $d \log_k n \in \Theta(\log_2 n)$.
Since $d$ is an arbitrary positive real number, $\forall d \in \mathbb{R}^+, d \log_k n \in \Theta(\log_2 n)$.
Hence $\forall k \in \mathbb{N}, k > 1 \Rightarrow \forall d \in \mathbb{R}^+, d \log_k n \in \Theta(\log_2 n)$.