The algebra of Θ

Serge Kruk

May 5, 2021

Three sets of functions

$$\Theta(g(n)) := \{f(n) \mid \exists c_1, c_2, n_0 > 0 : c_1 g(n) \leq f(n) \leq c_2 g(n)\ \forall n \geq n_0\}$$
$$\Omega(g(n)) := \{f(n) \mid \exists c_1, n_0 > 0 : 0 \leq c_1 g(n) \leq f(n)\ \forall n \geq n_0\}$$
$$O(g(n)) := \{f(n) \mid \exists c_2, n_0 > 0 : 0 \leq f(n) \leq c_2 g(n)\ \forall n \geq n_0\}$$

These are pronounced

- big omega (lower bound)
- big Oh (upper bound)
- big theta (lower and upper bound)

First consequence

$f(n) \in \Theta(g(n))$ if and only if $f(n) \in \Omega(g(n))$ and $f(n) \in O(g(n))$

Discuss

$$2n^3 - n^2 + n - 2 \in \Theta(n^3)$$

Notation

The book defines

$$f(n) = \Theta(g(n))$$

as

$$f(n) \in \Theta(g(n))$$

A useful notation, but always recall

- $f(n) = \Theta(g(n))$ does not imply $\Theta(g(n)) = f(n)$

Discuss

Consider the following function

```
def mystery(n):
    t = 0
    for i in range(1,n+1):
        s = 1
        for j in range(1,n+1):
            s = s * j
        t = t + s
    return t
```

- What does it return?
- What is its asymptotic runtime?

What does the algorithm return?

$$n * n!$$

Detailed analysis (and I mean detailed!)

$$
\begin{aligned}
T(n) &= 1 + \sum_{i=1}^{i=n} (2 + \sum_{j=1}^{j=n} 1) \\
&= 1 + \sum_{i=1}^{i=n} (2 + n) \\
&= 1 + n * (2 + n) \\
&= 1 + 2n + n^2
\end{aligned}
$$

- Longhand: This algorithm has a runtime proportional to $2n + n^2$
- Shorthand: This algorithm is $\Theta(n^2)$ (in the set $\Theta(n^2)$)

Discuss

True or False?

$$n^2 \in O(n^3)$$

True

We need c, n_0, both positive such that, for all $n \geq n_0$

$$n^2 \leq cn^3$$

which is equivalent to

$$1 \leq cn$$

Take $c = 1$ and $n_0 = 1$.

Discuss

True or False?

$$n^3 \in O(n^2)$$

False

We need c, n_0, both positive such that, for all $n \geq n_0$

$$n^3 \leq cn^2 \tag{1}$$

which is equivalent, for all $n \geq n_0$

$$n \leq c$$

Clearly nonsense.

Discuss

True or False?

$$2^{n+1} \in O(2^n)$$

True

Take $c = 2$.

In this course (in CS in general)

In textbooks, i.e. in introductory material, we only regularly see

$$O(1), O(\log n), O(n), O(n \log n), O(n^2), O(n^3), O(2^n), O(n!)$$

There are very few exceptions.

More algebra of Θ

Prove that for non-negative functions f, g,

$$h(n) := \max\{f(n), g(n)\} \in \Theta(f(n) + g(n))$$

More algebra of Θ

Prove that for non-negative functions f, g,

$$h(n) := \max\{f(n), g(n)\} \in \Theta(f(n) + g(n))$$

We need to exhibit c_1, c_2, n_0 such that

$$c_1(f(n) + g(n)) \leq h(n) \leq c_2(f(n) + g(n))$$

One inequality is straightforward

$$h(n) \leq f(n) + g(n)$$

Because the functions are non-negative.
So, take $c_2 = 1$ and $n_0 = 1$.

The other inequality needs a trick

Note that

$$f(n) \leq h(n)$$
$$g(n) \leq h(n)$$

Since h is the maximum. Now add the inequalities to get

$$\frac{1}{2}(f(n) + g(n)) \leq h(n)$$

And we can therefore take $c_1 = \frac{1}{2}$. $\square$

Homework/Test questions

- True or False $2n^3 - n \in O(n^3)$.
- True or False $2^{n-1} \in \Omega(2^n)$.
- True or False $O(n \log n) \subseteq O(\log n)$.
- True or False $O(\log n) \subseteq O(2^n)$.
- True or False $O(n^{3.1}) \subseteq O(n^3)$.
- True or False Bubble sort and Insertion sort have the same worst-case asymptotic runtime.
- **Prove** that

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