

Order

COMP 215

Recap of Mon.

- In analyzing algorithms we want to find the number of times some basic operation is performed as a function of input size.
- E.g.
 - $T(n) = 10n$
 - $T(n) = n^2$
- This example raises an interesting question:
 - for $n = 5$, $10n = 50$, $n^2 = 25$
 - for $n = 100$, $10n = 1,000$, $n^2 = 10,000$
- What input sizes do we care about?

Asymptotic Performance

- What about:
 - $T(n) = 1,000,000n$
 - vs. $T(n) = .01n^2$
- In this class, we are often interested in asymptotic performance - How efficient is the algorithm as the input size approaches infinity?

Some More Examples

- What about these two cases:
 - $T(n) = 5n$
 - $T(n) = n$
- What about these two:
 - $T(n) = n^2 + 2n + 3$
 - $T(n) = n^2$
- Notice that
 - $1000^2 + 2*1000 + 3 = 1002003$
 - $1000^2 = 1000000$

Order Notation

- Looking at asymptotic performance amounts to disregarding constants and lower order terms.
- This is where big-O, Ω , Θ , etc. come in.
- We will be grouping complexity functions* that are asymptotically similar.
 - e.g. n^2 , $2n^2 + 2$, and $n^2 + n$ are all in $\Theta(n^2)$
 - In other words they are order n^2 .
- This sort of analysis is often easier than calculating the exact complexity.

*Functions that map positive integers to non-negative reals.

Big-O

- $O(f(n))$ is the set of complexity functions $g(n)$ for which there exists some positive real constant c and some nonnegative integer N such that for all $n \geq N$,

$$g(n) \leq c \times f(n).$$

- In other words, $g(n)$ grows no faster than $f(n)$.
- Is $5n^2 + 2 \in O(n^2)$?
- Is $n^2 + n \in O(n^2)$?
- Is $n^2 + n \in O(2^{n^n} \times n!)$?
- Big-O provides an asymptotic upper bound.

Big Ω

- $\Omega(f(n))$ is the set of complexity functions $g(n)$ for which there exists some positive real constant c and some nonnegative integer N such that for all $n \geq N$,

$$g(n) \geq c \times f(n).$$

- $g(n)$ grows at least as fast as $f(n)$.
- $n^3 \in \Omega(n^2)$?
- $5n^2 + 2 \in \Omega(n^2)$?
- $n^2 \in \Omega(5n^2)$?
- Ω provides an asymptotic lower bound.

Big Ω

- What about showing $n^3 \notin \Omega(5n^4)$?

Θ

- For a given complexity function $f(n)$,

$$\Theta(f(n)) = O(f(n)) \cap \Omega(f(n)).$$

- I.e. $g(n)$ is in $\Theta(f(n))$ if $g(n) \in O(f(n))$ and $g(n) \in \Omega(f(n))$
- Is $5n^2 + 2 \in \Theta(n^2)$?

Specifying Complexity Classes

- It can be shown that order is symmetric:

$$g(n) \in \Theta(f(n)) \text{ if and only if } f(n) \in \Theta(g(n))$$

- There are disjoint complexity classes.
- Also note, it is just as correct to say:
 $n^2 + n + 3 \in \Theta(n^2 + 13n + 11)$ as it is to say
 $n^2 + n + 3 \in \Theta(n^2)$.
- We generally specify complexity classes by their smallest member.

Little o

- $o(f(n))$ is the set of complexity functions $g(n)$ that satisfy the following: for *every* positive real constant c there is some nonnegative integer N such that for all $n \geq N$,

$$g(n) \leq c \times f(n).$$

- In other words $g(n)$ grows more slowly than $f(n)$.
- Is $n \in o(n^2)$?

For the Sake of Completeness: Little ω

- $\omega(f(n))$ is the set of complexity functions $g(n)$ that satisfy the following: for *every* positive real constant c there is some nonnegative integer N such that for all $n \geq N$,

$$g(n) \geq c \times f(n).$$

- In other words $g(n)$ grows more quickly than $f(n)$.

The Relationship Between o , O and Ω

- If $g(n) \in o(f(n))$ then $g(n) \in O(f(n)) - \Omega(f(n))$.
- Two parts:
 - If $g(n) \in o(f(n))$ then $g(n) \in O(f(n))$.
- and:
 - If $g(n) \in o(f(n))$ then $g(n) \notin \Omega(f(n))$.

Check Out the Properties of Order on p. 37!

- Number 7 is a good one:

If $c \geq 0$, $d > 0$, $g(n) \in O(f(n))$, and $h(n) \in \Theta(f(n))$, then

$$c \times g(n) + d \times h(n) \in \Theta(f(n))$$

Recap

- One (dangerous) way to remember these things:

$$g(n) \in O(f(n)) \quad \approx \quad g(n) \leq f(n)$$

$$g(n) \in \Omega(f(n)) \quad \approx \quad g(n) \geq f(n)$$

$$g(n) \in \Theta(f(n)) \quad \approx \quad g(n) = f(n)$$

$$g(n) \in o(f(n)) \quad \approx \quad g(n) < f(n)$$

$$g(n) \in \omega(f(n)) \quad \approx \quad g(n) > f(n)$$

A Little Calculus

$$\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} = \left\{ \begin{array}{l} c \quad \text{implies } g(n) \in \Theta(f(n)) \text{ if } c > 0 \\ 0 \quad \text{implies } g(n) \in o(f(n)) \\ \infty \quad \text{implies } g(n) \in \omega(f(n)) \end{array} \right.$$

L'Hopital's Rule

- If $f(x)$ and $g(x)$ are both differentiable with derivatives $f'(x)$ and $g'(x)$, and if

$$\lim_{n \rightarrow \infty} f(x) = \lim_{n \rightarrow \infty} g(x) = \infty, \text{ then}$$

$$\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} = \lim_{n \rightarrow \infty} \frac{g'(n)}{f'(n)}$$

Exercises

- Show that $\lg n \in O(n \lg n)$. (using def. Of big- O)
- Show that $n^2 \notin O(n)$. (using def. Of big- O)
- Show that $n \log_2 n \in o(n^2)$. (using limits)

- (Remember that $\log_a n = \frac{\ln n}{\ln a}$.)

- (Also remember that $\frac{d}{dx} \ln x = \frac{1}{x}$.)