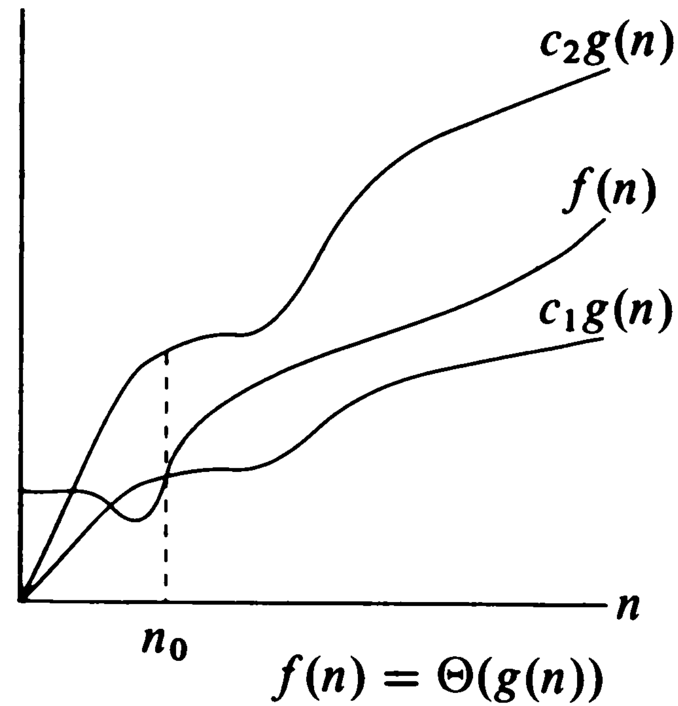
Asymptotic Analysis – Cont'd

Carlos Moreno
cmoreno@uwaterloo.ca
EIT-4103



https://ece.uwaterloo.ca/~cmoreno/ece250

These slides, the course material, and course web site are based on work by Douglas W. Harder

Asymptotic Analysis

Standard reminder to set phones to silent/vibrate mode, please!

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - We mentioned one example, for Θ(1) — a.k.a. constant (e.g., constant time) — being subscripted access of an element of an array; regardless the number of elements *n* in an array, accessing an element by subscript takes a constant amount of time (it maps to a single assembly-level instruction on most CPUs)

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - Perhaps a less obvious example for $\Theta(1)$ — an operation that may involve (perhaps at random), 1, 2, or 3 operations.
  - Why is this $\Theta(1)$?

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - Perhaps a less obvious example for Θ(1) — an operation that may involve (perhaps at random), 1, 2, or 3 operations.
  - Why is this Θ(1)?
  - Perhaps more strange is calling it *constant* time, when it is not constant at all.

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - You guys give me one example for logarithmic time ($\Theta(\log n)$)

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - How about linear time, or $\Theta(n)$?

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - How about linear time, or $\Theta(n)$?
  - Regular search on an array of n elements.
  - Computing the average of n values.
    - However, computing the variance would be .... ?

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - We saw an example for $n \log n$ — it was, however, $\Omega(n \log n)$, and not $\Theta(n \log n)$
  - Remember that one?

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - Later in the course, we'll see an example of sub-quadratic time.
  - A preview — an algorithm that recursively splits the job for size n into three recursive calls for size $n/2$.
    - Thus, if we do k recursive calls, that's 3^k operations (for each of the three recursive calls, there will be three other), but k will be $\lg n$ (right?), so that's $3^{\lg n}$ — from Q1 of the assignment, you recall that this is $n^{\lg 3} \approx n^{1.585}$

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - An example for quadratic time, $\Theta(n^2)$?
    - How about this for a trick question: can a sort algorithm be $\Theta(n^2)$?

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - In fact, let's see a (very simple/intuitive) sorting algorithm that is $\Theta(n^2)$
  - Find the lowest element in the array (that's linear time), then swap it with the first element; then, find the lowest element in the $n-1$ elements starting at the second position, and swap it with the second element; ... and so on.
  - BTW, this is known as selection sort — let's see why it is $\Theta(n^2)$

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - The searching part takes n operations the first time; then, $n-1$ operations, then $n-2$ .... all the way until just one operation.
  - For each of the first $n-1$ elements, you need to swap (for the last one, we don't need to do anything), which is 1 operation .... Well, ok, 3, but that's the same as 1 ... right? (why?)

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - We have, $n + (n-1) + (n-2) + \cdots + 3 + 2 + 1$ ... We recognize this .... (right?) ... And we have $3(n-1)$ operations for the swap — that's $n(n+1)/2 + 3(n-1)$, which is $\Theta(n^2)$

Asymptotic Analysis

- Examples of algorithms or operations exhibiting the common functions seen last time:
  - How about another trick question (maybe the question won't be too hard for you to see the answer right away, but I estimate that the answer may be very surprising)
  - What is the run time (in asymptotic notation) of selection sort when executed only in arrays of 5 elements or fewer? (we don't necessarily know how many, and it may vary between runs, but we know that it is always 5 or less elements)

Asymptotic Analysis

- And speaking of .... Let's bring back the example of linear search.
- We said that searching for a value in an unordered group of values requires $\Theta(n)$ operations — what if the value is found at the first, or second location?
- In fact, on average we will not require n operations — on average, it will be found half-way, so that would be .... (what exactly?)

Asymptotic Analysis

- Maybe we could suggest the idea that we want to describe linear search as $\mathrm{O}(n)$ instead of $\Theta(n)$ — big-Oh indicates upper-bound, instead of the tight bound that Θ describes.

Asymptotic Analysis

- But that's not really the point — where I'm going with this is the notion of how exactly we describe the run time of a given algorithm, with three important options to consider:
  - Worst-case
  - Average-case
  - Best-case

Asymptotic Analysis

- Worst-case analysis:
  Important when we need a guarantee about execution time.
  - For example, in Embedded Systems (in particular *real-time* embedded systems), this often plays an important role — example: if you're controlling a robot that is performing surgery, you have to calculate the movements, and once every, say, 10 ms, you absolutely need to output data to the actuators. You require the guarantee of execution time.

Asymptotic Analysis

- Average-case analysis:
  Quite often important — for most non-critical systems (e.g., games, web applications, server-side software, etc.); you want an average that allows you to guarantee a certain level of "smoothness" in the operation for a given number of concurrent requests.

Asymptotic Analysis

- Best-case analysis:
  Errmm.... Does *anyone* ever care about this one? :-)

Asymptotic Analysis

- Best-case analysis:
  Errmm.... Does *anyone* ever care about this one? :-)

  (yes, the implied sarcastic tone is because the answer is, «*of course no!*» Maybe as an academic exercise it may be interesting, perhaps for the purpose of making sure that we understand the complete picture, etc.; but it typically has no practical value)

Asymptotic Analysis

- In general, if we are restricted to one of these to describe a given algorithm, we use the worst-case to describe it.
- Often enough, we use both worst-case and average-case.
  - However, very often, in asymptotic notation, worst-case and average-case are in the same class (example, linear search — both worst-case and average-case are $\Theta(n)$)

Asymptotic Analysis

- Let's look at an alternative technique to figure out the appropriate descriptions for given functions.
- For example, consider two asymptotically non-negative functions f(*n*) and g(*n*), and consider:

$$\lim_{n \to \infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)}$$

- Say that the limit exists and is finite and non-zero — what does that tell us?

Asymptotic Analysis

- Well, you remember the formal definition of that limit ...

Asymptotic Analysis

- Well, you remember the formal definition of that limit ... (right?)

Asymptotic Analysis

- Well, you remember the formal definition of that limit ... (right?)

$$\lim_{n \to \infty} \mathrm{f}(n) = L \quad if$$

$$\forall\ \epsilon > 0,\ \ \exists\ N > 0 \ \Big|\ \left|f(n) - L\right| < \epsilon \ \ \forall\ n > N$$

Asymptotic Analysis

- Well, you remember the formal definition of that limit ... (right?)

$$\lim_{n \to \infty} \mathrm{f}(n) = L \quad if$$

$$\forall\ \epsilon > 0,\ \ \exists\ N > 0 \ \Big|\ \left| f(n) - L \right| < \epsilon \ \ \forall\ n > N$$

- Since we're saying that the limit of f(n) / g(n) exists (and is finite), we know that for all ε > 0, such *N* exists.

Asymptotic Analysis

- That is:

$$\forall\ \epsilon > 0,\ \ \exists\ N > 0\ \Bigg|\ \left|\frac{\mathrm{f}(n)}{\mathrm{g}(n)} - L\right| < \epsilon\ \ \forall\ n > N$$

- Since the above holds for every ε > 0, and *L* > 0 (why is *L* > 0?), then it must hold for every ε with 0 < ε < *L*:

$$\left|\frac{\mathrm{f}(n)}{\mathrm{g}(n)} - L\right| < \epsilon \ \Rightarrow\ -\epsilon < \frac{\mathrm{f}(n)}{\mathrm{g}(n)} - L < \epsilon$$

Asymptotic Analysis

- Thus,

$$-\epsilon < \frac{f(n)}{g(n)} - L < \epsilon \;\Rightarrow\; L-\epsilon < \frac{f(n)}{g(n)} < L+\epsilon$$

- Since g(*n*) is non-negative (right? why?):

$$(L-\epsilon)\,g(n) < f(n) < (L+\epsilon)\,g(n) \quad \forall\ n > N$$

- But $L-\varepsilon > 0$ (right?) ... So, what does this remind us of? (Hint: choose c1 = $L-\varepsilon$ and c2 = L+ε)

Asymptotic Analysis

- Bottom line:

  For every asymptotically non-negative functions f(*n*) and g(*n*), we have:

$$0 < \lim_{n \to \infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} < \infty \quad \Rightarrow \quad \mathrm{f}(n) = \Theta(\mathrm{g}(n))$$

- Important: this is a one-direction implication, and *not* an if-and-only-if — can you think of a counter-example (to show that the other direction does not necessarily hold)?

Asymptotic Analysis

- Similar criteria can be found for the other two Landau symbols that we saw last class:

$$\lim_{n \to \infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} < \infty \quad \Rightarrow \quad \mathrm{f}(n) = O(\mathrm{g}(n))$$

Asymptotic Analysis

- Similar criteria can be found for the other two Landau symbols that we saw last class:

$$\lim_{n \to \infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} < \infty \quad \Rightarrow \quad \mathrm{f}(n) = O(\mathrm{g}(n))$$

- The difference to notice being that now the limit may be 0 — and sure, f(n) = O(g(n)) allows for f(n) to be either asymptotically proportional to g(n) or asymptotically negligible with respect to g(n) — in which case the limit would be 0.

Asymptotic Analysis

- Similarly, for Ω we have:

$$0 < \lim_{n \to \infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} \quad \Rightarrow \quad \mathrm{f}(n) = \Omega(\mathrm{g}(n))$$

- The difference being that the limit may be ∞, corresponding to the situation where g(*n*) is asymptotically negligible with respect to f(*n*).

Asymptotic Analysis

- We can repeat the example (which will this time seem trivial by comparison) from last class, showing that $5n^2 + 2n = \Theta(n^2)$:

$$\lim_{n \to \infty} \frac{5n^2 + 2n}{n^2} = 5$$

- Directly showing that the numerator is Θ the denominator.

Asymptotic Analysis

- Another example: let's verify that selection sort is, as we know it must be, $\Omega(n \log n)$:

Asymptotic Analysis

- Another example: let's verify that selection sort is, as we know it must be, $\Omega(n \log n)$:

$$\lim_{n\to\infty} \frac{n^2}{n \log n} = \lim_{n\to\infty} \frac{n}{\log n}$$

Asymptotic Analysis

- Another example: let's verify that selection sort is, as we know it must be, $\Omega(n \log n)$:

$$\lim_{n\to\infty} \frac{n^2}{n \log n} = \lim_{n\to\infty} \frac{n}{\log n}$$

- We could claim that log grows slower than any power n^d, for every $d > 0$, and thus the limit is ∞.
- But we might as well use L'Hôpital's rule ...

Asymptotic Analysis

- Another example: let's verify that selection sort is, as we know it must be, $\Omega(n \log n)$:

$$\lim_{n \to \infty} \frac{n}{\log n} = \lim_{n \to \infty} \frac{1}{1/n} = \infty$$

Asymptotic Analysis

- Another example: let's verify that selection sort is, as we know it must be, $\Omega(n \log n)$:

$$\lim_{n \to \infty} \frac{n}{\log n} = \lim_{n \to \infty} \frac{1}{1/n} = \infty$$

- Perhaps the only warning is that n is not supposed to be a continuous variable .... (is this really a problem?)

Asymptotic Analysis

- There are two additional Landau symbols that we'll want to know about (though they're not as useful — in practical situations — for describing algorithms as the three that we have seen).
- These are little-oh (o), and little-omega (ω).
- You could visualize these as the part, or the subset, from their corresponding big counterparts that excludes the big-Theta.

Asymptotic Analysis

- For example, informally speaking, big-Oh means that the function is either proportional or negligible with respect to the other one.
  - Little-oh means that the function is negligible with respect to the other one.
- In terms of limits, it means:

$$\lim_{n \to \infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} = 0 \quad \Rightarrow \quad \mathrm{f}(n) = o(\mathrm{g}(n))$$

Asymptotic Analysis

- In terms of the formal definition, it may seem tricky/subtle:

$$o(\mathrm{g}(n)) = \left\{ \mathrm{f}(n) \;\middle|\; \begin{array}{l} \forall\ c>0,\ \ \exists\ \ N>0\ \ \textit{such that} \\ 0 \leqslant \mathrm{f}(n) \leqslant c\,\mathrm{g}(n)\ \ \forall\ n \geqslant N \end{array} \right\}$$

Asymptotic Analysis

- In terms of the formal definition, it may seem tricky/subtle:

$$o(\mathrm{g}(n)) = \left\{ \mathrm{f}(n) \;\middle|\; \begin{array}{l} \forall\ c>0,\ \ \exists\ \ N>0\ \ \textit{such that} \\ 0 \leqslant \mathrm{f}(n) \leqslant c\,\mathrm{g}(n)\ \ \forall\ n \geqslant N \end{array} \right\}$$

- See the difference with respect to big-Oh's definition?

Asymptotic Analysis

- Similar idea for little-omega — in terms of limits:

$$\lim_{n\to\infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} = \infty \quad \Rightarrow \quad \mathrm{f}(n) = \omega(\mathrm{g}(n))$$

Asymptotic Analysis

- Similar idea for little-omega — in terms of limits:

$$\lim_{n \to \infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} = \infty \quad \Rightarrow \quad \mathrm{f}(n) = \omega(\mathrm{g}(n))$$

- In terms of formal definition:

$$\omega(\mathrm{g}(n)) = \left\{ \mathrm{f}(n) \;\middle|\; \begin{array}{l} \forall\ c > 0,\ \exists\ N > 0 \ \ \textit{such that} \\ 0 \leqslant c\,\mathrm{g}(n) \leqslant \mathrm{f}(n)\ \ \forall\ n \geqslant N \end{array} \right\}$$

Asymptotic Analysis

- Summarizing the definitions, in terms of the criteria using limits:

$$\mathrm{f}(n) = o(\mathrm{g}(n)) \qquad \lim_{n\to\infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} = 0$$

$$\mathrm{f}(n) = O(\mathrm{g}(n)) \qquad \lim_{n\to\infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} < \infty$$

$$\mathrm{f}(n) = \Theta(\mathrm{g}(n)) \qquad 0 < \lim_{n\to\infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} < \infty$$

$$\mathrm{f}(n) = \Omega(\mathrm{g}(n)) \qquad \lim_{n\to\infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} > 0$$

$$\mathrm{f}(n) = \omega(\mathrm{g}(n)) \qquad \lim_{n\to\infty} \frac{\mathrm{f}(n)}{\mathrm{g}(n)} = \infty$$