Lecture 5: Growth of Functions

Agenda:

- Asymptotic notations $O, \Omega, \Theta, o, \omega$
- Growth of functions

Reading:

- Textbook pages 41 − 61

Motivations:

- Analysis of algorithms becomes analysis of functions:
  - *e.g.*,
    $f(n)$ denotes the WC running time of insertion sort
    $g(n)$ denotes the WC running time of merge sort
  - $f(n) = c_1 n^2 + c_2 n + c_3$
    $g(n) = c_4 n \log n$
  - Which algorithm is preferred (runs faster)?
- To simplify algorithm analysis, want function notation which indicates *rate of growth* (a.k.a., *order* of complexity)

$O(f(n))$ — read as "big O of $f(n)$"

roughly, The set of functions which, as n gets large, grow no faster than a constant times $f(n)$.

precisely, (or mathematically) The set of functions $\{h(n) : N \to R\}$ such that for each $h(n)$, there are constants $c_0 \in R^+$ and $n_0 \in N$ such that $h(n) \leq c_0 f(n)$ for all $n > n_0$.

examples: $h(n) = 3n^3 + 10n + 1000 \log n \in O(n^3)$

$h(n) = 3n^3 + 10n + 1000 \log n \in O(n^4)$

$$h(n) = \begin{cases} 5^n, & n \leq 10^{120} \\ n^2, & n > 10^{120} \end{cases} \in O(n^2)$$

Definitions:

- $O(f(n))$ is the set of functions $h(n)$ that
  - roughly, grow no faster than $f(n)$, namely
  - $\exists c_0, n_0$, such that $h(n) \leq c_0 f(n)$ for all $n \geq n_0$

- $\Omega(f(n))$ is the set of functions $h(n)$ that
  - roughly, grow at least as fast as $f(n)$, namely
  - $\exists c_0, n_0$, such that $h(n) \geq c_0 f(n)$ for all $n \geq n_0$

- $\Theta(f(n))$ is the set of functions $h(n)$ that
  - roughly, grow at the same rate as $f(n)$, namely
  - $\exists c_0, c_1, n_0$, such that $c_0 f(n) \leq h(n) \leq c_1 f(n)$ for all $n \geq n_0$
  - $\Theta(f(n)) = O(f(n)) \cap \Omega(f(n))$

- $o(f(n))$ is the set of functions $h(n)$ that
  - roughly, grow slower than $f(n)$, namely
  - $\lim_{n\to\infty} \frac{h(n)}{f(n)} = 0$

- $\omega(f(n))$ is the set of functions $h(n)$ that
  - roughly, grow faster than $f(n)$, namely
  - $\lim_{n\to\infty} \frac{h(n)}{f(n)} = \infty$
  - $h(n) \in \omega(f(n))$ if and only if $f(n) \in o(h(n))$

Warning:

- the textbook overloads "="
  - Textbook uses $g(n) = O(f(n))$
  - *Incorrect* !!!
    Because $O(f(n))$ is a *set* of functions.
  - Correct: $g(n) \in O(f(n))$
  - You should use the correct notations.

Examples: which of the following belongs to $O(n^3)$, $\Omega(n^3)$, $\Theta(n^3)$, $o(n^3)$, $\omega(n^3)$?

1. $f_1(n) = 19n$
2. $f_2(n) = 77n^2$
3. $f_3(n) = 6n^3 + n^2 \log n$
4. $f_4(n) = 11n^4$

Answers:

1. $f_1(n) = 19n$
2. $f_2(n) = 77n^2$
3. $f_3(n) = 6n^3 + n^2 \log n$
4. $f_4(n) = 11n^4$

- $f_1, f_2, f_3 \in O(n^3)$
  $f_1(n) \leq 19n^3$, for all $n \geq 0$ — $c_0 = 19$, $n_0 = 0$
  $f_2(n) \leq 77n^3$, for all $n \geq 0$ — $c_0 = 77$, $n_0 = 0$
  $f_3(n) \leq 6n^3 + n^2 \cdot n$, for all $n \geq 1$, since $\log n \leq n$
  if $f_4(n) \leq c_0 n^3$, then $n \leq \frac{c_0}{11}$ — no such n_0 exists

- $f_3, f_4 \in \Omega(n^3)$
  $f_3(n) \geq 6n^3$, for all $n \geq 1$, since $n^2 \log n \geq 0$
  $f_4(n) \geq 11n^3$, for all $n \geq 0$

- $f_3 \in \Theta(n^3)$
  why?

- $f_1, f_2 \in o(n^3)$
  $f_1(n)$: $\lim_{n\to\infty} \frac{19n}{n^3} = \lim_{n\to\infty} \frac{19}{n^2} = 0$
  $f_2(n)$: $\lim_{n\to\infty} \frac{77n^2}{n^3} = \lim_{n\to\infty} \frac{77}{n} = 0$
  $f_3(n)$: $\lim_{n\to\infty} \frac{6n^3 + n^2 \log n}{n^3} = \lim_{n\to\infty} 6 + \frac{\log n}{n} = 6$
  $f_4(n)$: $\lim_{n\to\infty} \frac{11n^4}{n^3} = \lim_{n\to\infty} 11n = \infty$

- $f_4 \in \omega(n^3)$

logarithm review:

- Definition of $\log_b n$ $(b, n > 0)$: $b^{\log_b n} = n$
- $\log_b n$ as a function in n: increasing, one-to-one
- $\log_b 1 = 0$
- $\log_b x^p = p \log_b x$
- $\log_b(xy) = \log_b x + \log_b y$
- $x^{\log_b y} = y^{\log_b x}$
- $\log_b x = (\log_b c)(\log_c x)$

Some notes on logarithm:

- $\ln n = \log_e n$ (natural logarithm)
- $\lg n = \log_2 n$ (base 2, binary)
- $\Theta(\log_b n) = \Theta(\log_{\{\text{whatever positive}\}} n) = \Theta(\log n)$
- $\frac{d}{dx} \ln x = \frac{1}{x}$
- $(\log n)^k \in o(n^\epsilon)$, for any positives k and ϵ

Handy 'big O' tips:

- $h(n) \in O(f(n))$ if and only if $f(n) \in \Omega(h(n))$
- limit rules: $\lim_{n\to\infty} \frac{h(n)}{f(n)} = \ldots$
  - $\ldots\infty$, then $h \in \Omega(f), \omega(f)$
  - $\ldots 0 < k < \infty$, then $h \in \Theta(f)$
  - $\ldots 0$, then $h \in O(f), o(f)$
- L'Hôspital's rules: if $\lim_{n\to\infty} h(n) = \infty$, $\lim_{n\to\infty} f(n) = \infty$, and $h'(n)$, $f'(n)$ exist, then

  $$\lim_{n\to\infty} \frac{h(n)}{f(n)} = \lim_{n\to\infty} \frac{h'(n)}{f'(n)}$$

  e.g., $\lim_{n\to\infty} \frac{\ln n}{n} = \lim_{n\to\infty} \frac{1}{n} = 0$
- Cannot always use L'Hôspital's rules. *e.g.*,
  - $h(n) = \begin{cases} 1, & \text{if } n \text{ even} \\ n^2, & \text{if } n \text{ odd} \end{cases}$
  - $\lim_{n\to\infty} \frac{h(n)}{n^2}$ does NOT exist
  - Still, we have $h(n) \in O(n^2)$, $h(n) \in \Omega(1)$, etc.
- $O(\cdot), \Omega(\cdot), \Theta(\cdot), o(\cdot), \omega(\cdot)$

  JUST useful asymptotic notations

Have you understood the lecture contents?

well	ok	not-at-all	topic
☐	☐	☐	definitions: $O, \Omega, \Theta, o, \omega$
☐	☐	☐	how to prove $h(n) \in O(f(n))$
☐	☐	☐	logarithm
☐	☐	☐	use of L'Hôspital's rules

Question #4:

Five distinct elements are randomly chosen from integers between 1 and 20, and stored in a list $L[1]$, ..., $L[5]$. Using linear search we want to determine if an integer x (also chosen randomly from integers between 1 and 20) belongs to the list L.

1. What is the number of *key* comparisons required on the average?

2. Give a similar analysis as in the first part if L has n elements and all numbers are selected from integers between 1 and m.

Hints:

- The probability that you need exactly 1 comparison is $\frac{1}{20}$, because x is randomly chosen and thus it hits the first number with that probability.

- What about 2 comparisons?

  Still $\frac{1}{20}$. Why?

- What about 3 comparisons then?

- Sum them up:

$$\frac{1}{20} \times 1 + \frac{1}{20} \times 2 + \frac{1}{20} \times 3 + \frac{1}{20} \times 4 + \frac{20-4}{20} \times 5 = \frac{90}{20} = 4.5$$

- For the general question, do the same analysis and the answer is $\frac{2mn-n^2+n}{2m}$.