CS3230 Semester 1 2024/2025
Design and Analysis of Algorithms

Tutorial 01
Introduction and Asymptotic Analysis
For Week 02

Document is last modified on: August 17, 2024

1 Notes

CS3230 tutorial format is as follows: We will consider a few questions per tutorial. Some questions are **revealed beforehand** (published at `https://www.comp.nus.edu.sg/~stevenha/cs3230.html`), some are **hidden** (usually a variation of the public version) and will only be discussed on the spot.

For **each question**, we will ask a student to solve it. A **reasonable** attempt for that question will earn the student one participation point (1%). The **limit is maximum 3 points (3%)** for a student for the whole semester. TA will try to ensure that each student do at least one question throughout the semester. As there are 11 tutorials and $\approx$ [5..6] questions per tutorial, we have computed that each student should get $\frac{11 \times 5.5}{27} \approx [2..2.5]\%$ participation points.

Note that since this is the first tutorial, your TA will start the session with a short icebreaker.

2 Lecture Review: Asymptotic Analysis

We say[1] $f \in O(g)$ or $f(n) \in O(g)$ or $f = O(g)$ or $f(n) = O(g(n))$ if $\exists c, n_0 > 0$ such that $\forall n \geq n_0, 0 \leq f(n) \leq c \cdot g(n)$. Informally, we say (function) g is an upper bound on (function) f. This is the most popular Big O worst-case time complexity analysis that we have learned since earlier courses, i.e., from CS2040/C/S.

Copy-pasting similar mathematical statement four other times for the other asymptotic notations $\Omega, \Theta, o, \omega$ is probably less clear compared to the following tabular summary:

[1] FAQ: We are fine with either notation although we prefer $f(n) \in O(g(n))$ notation. It is a massive editing exercise to change all occurrences of $f(n) = O(g(n))$ into $f(n) \in O(g(n))$ from the previous semesters, so forgive us if some notations were still the 'less preferred' ones.

We say	if $\exists c, c_1, c_2, n_0 > 0$ such that $\forall n \geq n_0$	In other words
$f(n) \in O(g(n))$	$0 \leq f(n) \leq c \cdot g(n)$	g is an **upper** bound on f
$f(n) \in \Omega(g(n))$	$0 \leq c \cdot g(n) \leq f(n)$	g is a **lower** bound on f
$f(n) \in \Theta(g(n))$	$0 \leq c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$	g is a **tight** bound on f

We say	$\forall c > 0$, $\exists n_0 > 0$ such that $\forall n \geq n_0$	In other words
$f(n) \in o(g(n))$	$0 \leq f(n) < c \cdot g(n)$	g is a **strict upper** bound on f
$f(n) \in \omega(g(n))$	$0 \leq c \cdot g(n) < f(n)$	g is a **strict lower** bound on f

3 Tutorial 01 Questions

Q1). Assume $f(n), g(n) > 0$, show:

- $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0 \Rightarrow f(n) \in o(g(n))$ — this has already been shown in lec01b.
- $\lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty \Rightarrow f(n) \in O(g(n))$
- $0 < \lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty \Rightarrow f(n) \in \Theta(g(n))$
- $\lim_{n\to\infty} \frac{f(n)}{g(n)} > 0 \Rightarrow f(n) \in \Omega(g(n))$
- $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty \Rightarrow f(n) \in \omega(g(n))$

Q2). Assume $f(n), g(n) > 0$, show:

- Reflexivity
  - $f(n) \in O(f(n))$
  - $f(n) \in \Omega(f(n))$
  - $f(n) \in \Theta(f(n))$
- Transitivity
  - $f(n) \in O(g(n))$ and $g(n) \in O(h(n)$ implies $f(n) \in O(h(n))$
  - Do the same for Ω, Θ, o, ω
- Symmetry
  - $f(n) \in \Theta(g(n))$ iff $g(n) \in \Theta(f(n))$
- Complementarity
  - $f(n) \in O(g(n))$ iff $g(n) \in \Omega(f(n))$
  - $f(n) \in o(g(n))$ iff $g(n) \in \omega(f(n))$

Q3). Which of the following statement(s) is/are True?

1. $3^{n+1} \in O(3^n)$
2. $4^n \in O(2^n)$
3. $2^{\lfloor \log n \rfloor} \in \Theta(n)$ (we assume log is in base 2)
4. For a constant $i, a > 0$, we have $(n+a)^i \in O(n^i)$

Q4). Which of the following statement(s) is/are True?
$2^{\log_2 n} \in$

1. $O(n)$
2. $\Omega(n)$
3. $\Theta(\sqrt{n})$
4. $\omega(n)$

Q5). Rank the following functions by their order of growth.
(But if any two (or more) functions have the same order of growth, group them together).

- $f_1(n) = \log n$
- $f_2(n) = n!$
- $f_3(n) = 2^n + n$
- $f_4(n) = n^{2.3} + 16n$