# CS 624: Notes 03

Ryan Culpepper

September 14, 2022

*These notes contain an outline of what I said in lecture (but only an outline), and they also contain interactive questions and exercises. The corresponding slides are in* `slides02.pdf`.

## 1 Administrative

- Homework 01 is out.
- Join Piazza.
- Any problems with previous lecture recordings?

  no

## 2 From last time

We ended last lecture with an analysis of the running time of Merge Sort. We calculated the time with a full tree of height $(\log_2 n)$, even though some branches will reach the base case sooner than others (if $n \neq 2^k$). Why doesn't that matter?

Rationale: Since we have already abstracted away machine details, we have committed to not caring about *constant factors*.

In other words, trying to precisely count the number of "steps" an algorithm will take is useless, since different steps can take different amounts of time anyway.

1. If we ignore constant factors, do any differences actually remain? If so, what run times are actually different?
2. How do we rigorously define "different even disregarding constant factors"? This is called *asymptotic** efficiency/behavior/etc.
3. What techniques can we use to calculate or prove the asymptotic efficiency of an algorithm?

# 3 Order of growth

(slides 2-5)

# 4 Asymptotic Notation

(slides 6-14)

It is common to write things like

$$f(n) = \text{runtime of InsertionSort on array of length } n$$
$$f(n) = O(n^2)$$

Two notational abuses:

- $O(n^2)$ means $O(\{n \mapsto n^2\})$ or $O(\lambda n.n^2)$ or $O(g)$ where $g(n) = n^2$
  The argument to $O$, $\Theta$, $\Omega$, etc is a *function*. But when there is a clear variable in question, we'll talk about an expression as if it were an implicit function.
- $f(n) = O(n^2)$ means $f \in O(\{n \mapsto n^2\})$
  $O(n^2)$ (or $O(\{n \mapsto n^2\})$) is not a function, it is a *set of functions*. But we'll write "=" instead of "$\in$", and in some cases we'll write $O(n^2)$ to mean some (unknown or unspecified) function in that set.

We assume that all functions are *asymptotically nonnegative*.

Why is $x = O(x^2)$?

> $c = 1$
> $x_0 = 1$ Then for every $x \geq 1$:
> $x \leq 1x^2$

Is $10n = O(n^2)$?

> Yes. Why?
> $c = 1$
> $n_0 = 10$
>
> When is $10 \times n \leq n \times n$?
> When $10 \leq n$.
>
> Another solution: $c = 10$
> $n_0 = 1$

Is $5n^2 = O(n^{2.001})$?

> $c = 5$
> $n_0 = 1$

$5 \times n^2 \leq c \times n^2 \times n^{0.001}$
when $5 \leq c \times n^{0.001}$

(stopped at slide 20) (Homework 01 deadline will be extended)