Week 1 - Friday

COMP 4500

Last time

- Finished Big Theta examples
- Proof techniques
- Tractability

Questions?

Logical warmup

- For Diwali, Mr. Patel's five daughters gave each other books as presents.
- Each presented four books and each received four books, but no two girls divided her books in the same way.
- That is, only one gave two books to one sister and two to another. **Bharat** gave all her books to **Abhilasha**; **Chandra** gave three to **Esha**.
- Who gave how many books to whom? (Mr. Patel's fourth daughter is named **Deeti**.)

Proving Universal Statements

A useful definition

- We'll start with basic definitions of even and odd to allow us to prove simple theorems
- If n is an integer, then:
  - n is even $\Leftrightarrow \exists\, k \in \mathbf{Z}$ such that $n = 2k$
  - n is odd $\Leftrightarrow \exists\, k \in \mathbf{Z}$ such that $n = 2k + 1$
- Since these are bidirectional, each side implies the other

Generalizing from the generic particular

- Pick some specific (but arbitrary) element from the domain
- Show that the property holds for that element, just because of that properties that any such element must have
- Thus, it must be true for all elements with the property
- Example: $\forall \boldsymbol{x} \in \mathbf{Z}$, if $\boldsymbol{x}$ is even, then $\boldsymbol{x} + 1$ is odd

Direct proof

- Direct proof uses the method of generalizing from a generic particular, following these steps:
  1. Express the statement to be proved in the form $\forall \boldsymbol{x} \in \boldsymbol{D}$, if $\boldsymbol{P}(\boldsymbol{x})$ then $\boldsymbol{Q}(\boldsymbol{x})$
  2. Suppose that $\boldsymbol{x}$ is some specific (but arbitrarily chosen) element of $\boldsymbol{D}$ for which $\boldsymbol{P}(\boldsymbol{x})$ is true
  3. Show that the conclusion $\boldsymbol{Q}(\boldsymbol{x})$ is true by using definitions, other theorems, and the rules for logical inference

Direct proof example

- Prove the sum of any two odd integers is even.

Proof by contradiction

- In a proof by contradiction, you begin by assuming the **negation** of the conclusion
- Then, you show that doing so leads to a logical impossibility
- Thus, the assumption must be false and the conclusion true

Contradiction formatting

- A proof by contradiction is different from a direct proof because you are **trying** to get to a point where things don't make sense
- You should always clearly state that it's a **proof by contradiction**
- You will reach a point where you have ***p*** and ~***p***, mark that as a **contradiction**
- If you're doing a proof by contradiction and you actually show the thing you wanted to prove in the first place, **it's not a proof!**

Proof by contradiction example

- **Theorem:** There is no integer that is both even and odd.
- **Proof by contradiction:** Assume that there is an integer that is both even and odd

$\sqrt{2}$ is irrational

Theorem: $\sqrt{2}$ is irrational

Proof by contradiction:

	Statement		Reason
1.	Suppose $\sqrt{2}$ is rational	1.	Negation of conclusion
2.	$\sqrt{2} = \boldsymbol{m/n}$, where $\boldsymbol{m, n} \in \mathbf{Z}$, $\boldsymbol{n} \neq 0$ and $\boldsymbol{m}$ and $\boldsymbol{n}$ have no common factors	2.	Definition of rational
3.	$2 = \boldsymbol{m}^2/\boldsymbol{n}^2$	3.	Squaring both sides
4.	$2\boldsymbol{n}^2 = \boldsymbol{m}^2$	4.	Multiply both sides by $\boldsymbol{n}^2$
5.	$2\boldsymbol{k} = \boldsymbol{m}^2$, $\boldsymbol{k} \in \mathbf{Z}$	5.	Square of integer is integer
6.	$\boldsymbol{m} = 2\boldsymbol{a}$, $\boldsymbol{a} \in \mathbf{Z}$	6.	Even $\boldsymbol{x}^2$ implies even $\boldsymbol{x}$ (Proven elsewhere)
7.	$2\boldsymbol{n}^2 = (2\boldsymbol{a})^2 = 4\boldsymbol{a}^2$	7.	Substitution
8.	$\boldsymbol{n}^2 = 2\boldsymbol{a}^2$	8.	Transitivity
9.	$\boldsymbol{n} = 2\boldsymbol{b}$, $\boldsymbol{b} \in \mathbf{Z}$	9.	Even $\boldsymbol{x}^2$ implies even $\boldsymbol{x}$
10.	2 divides ***m*** and 2 divides ***n***	10.	Conjunction of 6 and 9, **contradiction**
11.	$\sqrt{2}$ is irrational	11.	By contradiction in 10, supposition is false

■

Asymptotic Order of Growth

Rule of thumb

- We want a way to bound the size of an algorithm's running time in terms of its input size
- Measuring the exact number of operations would be a lot of detailed work
  - Which would probably be invalid in a different programming language or on a different processor
- Instead, a simplified, rough outline of the speed at which a running time increases with input size is more useful

Upper bounds

- Let $T(n)$ be the running time of an algorithm
- Let $f(n)$ be a non-decreasing function
- $T(n)$ is $O(f(n))$ if and only if
  - $T(n) \leq c \cdot f(n)$ for all $n \geq n_0$ where $n_0 \geq 0$
  - for **some** positive real numbers c and n_0
- In other words, past some arbitrary point, with some arbitrary scaling factor, $f(n)$ is at least as big
- We say that $T(n)$ is **upper bounded** by $f(n)$

Lower order terms don't matter

- Why not?
- We can assume that $n \geq 1$
- In that situation, $n \leq n^2 \leq n^3 \leq n^4$, etc.
- Thus, they can get wrapped into our constant:
  - $pn^2 + qn + r \leq pn^2 + qn^2 + rn^2 = (p + q + r)n^2$
- From a practical perspective, lower order terms will also have relatively no impact when n gets large

Lower bounds

- Let $T(n)$ be the running time of an algorithm
- Let $f(n)$ be a non-decreasing function
- $T(n)$ is $\Omega(f(n))$ if and only if
  - $T(n) \geq \varepsilon \cdot f(n)$ for all $n \geq n_0$ where $n_0 \geq 0$
  - for **some** positive real numbers ε and n_0
- In other words, past some arbitrary point, with some arbitrary scaling factor, $f(n)$ is no bigger
- We say that $T(n)$ is **lower bounded** by $f(n)$

Tight bounds

- Let $T(n)$ be the running time of an algorithm
- Let $f(n)$ be a non-decreasing function
- If $T(n)$ is $O(f(n))$ and $\Omega(f(n))$, we say that $T(n)$ is $\Theta(f(n))$
- In other words, past some arbitrary point, with some arbitrary scaling factor, $f(n)$ grows at about the same rate
- We say that $T(n)$ is **tightly bounded** by $f(n)$

Another way to look at tight bounds

- Given two functions $\boldsymbol{f}(\boldsymbol{n})$ and $\boldsymbol{g}(\boldsymbol{n})$, if

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

- Then $\boldsymbol{f}(\boldsymbol{n})$ is is $\Theta(\boldsymbol{g}(\boldsymbol{n}))$ (and vice versa)
- Why?
  - Because of how a limit works, there is some $\boldsymbol{n}$ beyond which the ratio of $\boldsymbol{f}(\boldsymbol{n})$ to $\boldsymbol{g}(\boldsymbol{n})$ will be between $\frac{1}{2}c$ and $2c$, making $\boldsymbol{f}(\boldsymbol{n})$ both $\boldsymbol{O}(\boldsymbol{g}(\boldsymbol{n}))$ and $\Omega(\boldsymbol{g}(\boldsymbol{n}))$

Abuse of notation

- Both this book and many others "abuse" notation by saying things like:
  - $T(n) = O(n^2)$
  - $T(n) = \Omega(\log n)$
  - $T(n) = \Theta(\sqrt{n})$
- Those equal signs do **not** represent mathematical equality
- Instead, they should be read **"is"**
- It's a shorthand
- I recommend that you do **not** use it

Properties of Asymptotic Bounds

Transitivity

- If $f(n)$ is $O(g(n))$ and $g(n)$ is $O(h(n))$, then $f(n)$ is $O(h(n))$
- If $f(n)$ is $\Omega(g(n))$ and $g(n)$ is $\Omega(h(n))$, then $f(n)$ is $\Omega(h(n))$
- If $f(n)$ is $\Theta(g(n))$ and $g(n)$ is $\Theta(h(n))$, then $f(n)$ is $\Theta(h(n))$
- Prove it.

Sums

- If $f(n)$ is $O(h(n))$ and $g(n)$ is $O(h(n))$, then $f(n) + g(n)$ is $O(h(n))$
- Prove it.
- If $g(n)$ is $O(f(n))$, then $f(n) + g(n)$ is $\Theta(f(n))$
  - This is another way of showing that lower order terms don't matter.

Polynomial bounds

- A polynomial of degree d can be written as $f(n) = a_0 + a_1 n + a_2 n^2 + \ldots + a_d n^d$, where $a_d > 0$
- For any such polynomial, $f(n)$ is $\Theta(n^d)$
- To prove this, note that any term $a_j n^j \leq |a_j| n^d$ when $n > 1$

Logarithms

- There is a function called the logarithm with base ***b*** of ***x*** defined from $\mathbf{R}^+$ to $\mathbf{R}$ as follows:
  - $\log_b x = y \Leftrightarrow b^y = x$
- Logarithms are **very** slowly growing functions, slower than **any** polynomial function:
  - For any $b > 1$ and every $x > 0$, $\log_b n$ is $O(n^x)$
  - Even $n^{0.00001}$ grows faster than $\log n$

Bases don't matter

- The base of a logarithm doesn't matter in asymptotic notation
- Why?
- $\log_a n = \dfrac{\log_b n}{\log_b a}$
- Thus, $\log_b \boldsymbol{n} = (\log_b \boldsymbol{a})\log_a \boldsymbol{n} = \boldsymbol{c}\log_a \boldsymbol{n}$

Exponential bounds

- On the other end of the spectrum, any exponential with a base $r > 1$ will grow faster than any polynomial
  - For every $r > 1$ and every $d > 0$, n^d is $O(r^n)$
  - Even 1.0001^n grows faster than n^{1000}
- People talk about "exponential time," but all exponents are actually different
  - For every $r > 1$ and every $s > r$, r^n is $O(s^n)$

Three-sentence Summary of Stable Marriage and Five Representative Problems

Stable Marriage

Imagine *n* men and *n* women

- All 2***n*** people want to get married
- All of them are *willing* to marry any of the n members of the opposite gender
- Each woman has ranked all ***n*** men in order of preference
- Each man has ranked all ***n*** women in order of preference
- We want to match them up so that the marriages are **stable**

Stability

- Consider two marriages:
  - Anna and Bob
  - Caitlin and Dan
- This pair of marriages is unstable if
  - Anna likes Dan more than Bob **and** Dan likes Anna more than Caitlin **or**
  - Caitlin likes Bob more than Dan **and** Bob likes Caitlin more than Anna
- We want to arrange all ***n*** marriages such that none are unstable

Upcoming

Next time…

- Finish stable marriage
- Five representative problems
- Implementing stable marriage

Reminders

- **No class Monday!**
- Read Section 2.3
- Assignment 1 is due next Friday