# Introduction to Algorithms

**William Hendrix**

*Lecture 1*

# Syllabus

- Read at: usflearn.instructure.com

- Contact: ENB 344F or whendrix@usf.edu
- Office hours: Immediately after class
- TAs: Mehrad Eslami and Martin Llofriu
- Office hours: TBD

- Feedback forms
- Course objectives, grading scale, exams, etc.
- Class participation
- Attendance policy
- Piazza

# What will we learn in this class?

- **How to determine if an algorithm is correct**
  - Proofs of correctness
  - Loop invariants
- **How to determine if an algorithm is efficient**
  - RAM model
  - Big-Oh definition and properties
- **How to develop your own algorithms**
  - Greedy algorithms
  - Divide-and-conquer
  - Dynamic programming
  - Graph traversals
  - Backtracking
- **Whether you should develop your own algorithm**
  - Canonical algorithms
  - Complexity theory

# A Word of Warning

- Algorithms is not an easy class

- It requires:
  - Strong mathematical background
    - Writing proofs
    - Understanding graphs
  - An understanding of data structures
    - Arrays, lists, trees, graphs
  - Creativity!
  - Good work ethic and academic integrity

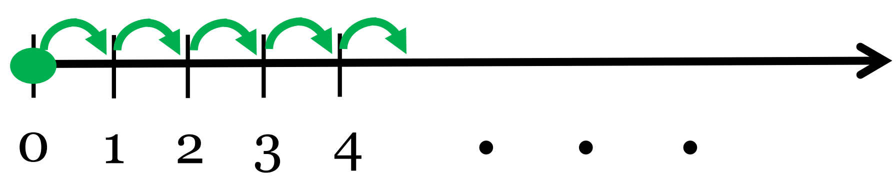
- The projects require:
  - Strong programming skills
    - C, C++, or Java

# Resources

- Read the textbook
- Ask questions on Piazza
- Visit office hours
- Solve practice problems
  - From the textbook or the web
- Contact me via email or Canvas

# Proof review

- Induction
  - Proof technique for statements of the form "$P(i)$, for all $i \geq b$."
  - Two parts (<u>both required</u>)
    - Base case: prove $P(b)$
    - Inductive step: assume $P(k)$ for some $k \geq b$
    - Strong induction: assume $P(x)$ for all $x$ from $b$ up to $k$
    - Prove $P(k+1)$




- Contradiction
  - Assume claim is false
  - Show something impossible must be true
- Example/counterexample
  - Used to prove existence/nonexistence
  - Construct an example and show it meets the criteria

# Induction exercise

Prove that $n! > 2^{n+1}$ for all $n \geq 5$.

# Induction solution

Prove that $n! > 2^{n+1}$ for all $n \geq 5$.

*Proof.* (*Base case*) $5! = 120$, and $2^{5+1} = 2^6 = 64$. Since $120 > 64$, $n! > 2^{n+1}$ for $n = 5$.
(*Inductive step*) Suppose that $k! > 2^{k+1}$, for some $k \geq 5$, and consider $n = k + 1$. $n! = (k+1)! = (k+1)k!$, while $2^{n+1} = 2^{k+2} = 2(2^{k+1})$.

$$
\begin{aligned}
n! &= (k+1)k! && (1)\\
&> (k+1)2^{k+1} && (2)\\
&> 2(2^{k+1}) && (3)\\
&= 2^{n+1} && (4)
\end{aligned}
$$

(Note: in line 3, $k+1 > 2$ because $k \geq 5$.) Since $k! > 2^{k+1}$ implies that $(k+1)! > 2^{k+2}$, $n! > 2^{n+2}$ for all $n \geq 5$, by induction.

$\square$

# Strong induction exercise

- Consider a two-player game where stones are arranged into piles. On their turn, a player chooses a pile with two or more stones and divides it into two piles, each with at least one stone. The game ends when all of the piles only have one stone.
- Prove the following:
  - A pile with $n$ stones will always take $n - 1$ turns to split up into $n$ piles of one stone, for all $n \geq 1$.

# Strong induction solution

- A pile with $n$ stones will always take $n - 1$ turns to split up into $n$ piles of one stone, for all $n \geq 1$.

*Proof.* (*Base case*) When $n = 1$, the pile has 1 stone, and no moves are required to split it into piles of size 1, so it takes $0 = n - 1$ moves.

(*Inductive step*) Suppose that all piles of $x$ stones take $x - 1$ moves to divide into piles of size 1, for any $x \leq k$.

Consider a pile with $k + 1$ stones. On their first move, a player will divide this pile into piles of size $a$ and $b$, where $a$ and $b$ are both at least one and $a + b = k + 1$. Since $a$ and $b$ are both at least one and $a + b = k + 1$, they can't be any larger than $k$, so the inductive hypothesis applies. Hence, the pile with $a$ stones will take $a - 1$ moves to fully split, and the pile with $b$ stones will take $b - 1$ moves to fully split. Thus, the total number of moves to split $k + 1$ is $1 + (a - 1) + (b - 1) = a + b - 1 = (k + 1) - 1 = k$. Therefore, by induction, all piles with $n$ stones will take $n - 1$ moves to fully split. $\square$

# Contradiction exercise

- Prove that if $n$ is composite, $n$ must have a factor no larger than $\sqrt{n}$.

# Contradiction solution

- Prove that if $n$ is composite, $n$ must have a factor no larger than $\sqrt{n}$.

*Proof.* We prove the claim by contradiction.

Suppose $n$ is a composite number that does not have a factor less than or equal to $\sqrt{n}$. Since $n$ is composite, $n = ab$, for some integers $a, b > 1$. Since $a$ and $b$ are factors of $n$, they must be larger than $\sqrt{n}$, so $ab > \sqrt{n}\sqrt{n} = n$. However, it's not possible for $ab > n$, since $n = ab$. $\Rightarrow\Leftarrow$

Hence, every composite number must have a factor no larger than its square root. $\square$

# Counterexample exercise

- **Disprove** the claim that if $b$ and $p$ are positive integers such that $b$ is not a multiple of $p$, then $b^{p-1}$ must be 1 more than a multiple of $p$.

# Counterexample solution

- **Disprove** the claim that if $b$ and $p$ are positive integers such that $b$ is not a multiple of $p$, then $b^{p-1}$ must be 1 more than a multiple of $p$.

Any example where $b$ and $p$ have a factor in common will fail. E.g., when $p = 6$ and $b = 3$, $b^{p-1} = 3^5 = 243$, which has remainder of 3 mod 6, not 1.

# Coming up

- Essential definitions
- Proving correctness
- Loop invariants

- Sign today's **sign-in sheet**!
- **Homework 1** will be posted later this evening
  – Due next Tuesday
- Feedback form 1 is posted on Canvas
  – Due at Exam 1

- **Recommended readings:** Section 2.1
- **Practice Problems:** 2.1-3