Middlebury

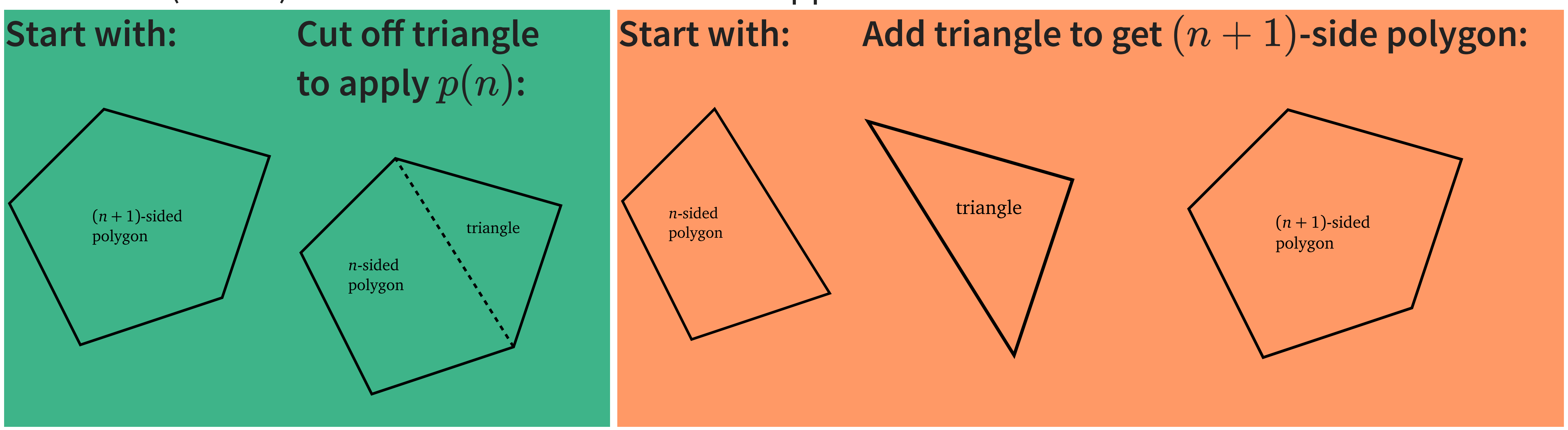
CSCI 200: Math Foundations of Computing

Spring 2026

Lecture 10M: Counting

A few notes about Midterm 2.

- I.H. is a predicate: incorrect to use quantifiers, incorrect to set $p(n)$ equal to a number.
- In the inductive step: (P2) starting with an ℓ-leaf node full binary tree and adding two leaf nodes; (P4) starting with an n-sided polygon and building an $(n+1)$-sided polygon.
  - Prove, using induction, that the total number of nodes (n) in a full binary tree is ...
  - Prove using induction that the sum of the internal angles of a convex polygon with n sides is ...
  - There IS a (subtle) difference between these two approaches:



  - HOWEVER This was not graded incorrectly because of how I accidentally worded the problems ("every"/"any" versus "a").
- P3: manipulating both the LHS and RHS to find an equivalent expression.
- P5: Not going through the BFS steps to build the spanning tree. It's fine to stop once the destination node is reached, but the spanning tree and order in which nodes are visited before reaching the destination needs to be shown. Otherwise, there is no argument for why this is the shortest path (BFS is incomplete) and resembles DFS instead.

Goals for today:

1. Relate properties of functions to the **mapping rule**,
2. Apply **product** and **addition rules** to **count sets** without explicitly listing all items in the set,
3. Apply the **pigeonhole principle** to relate the cardinalities of two sets.

In what month is your birthday?

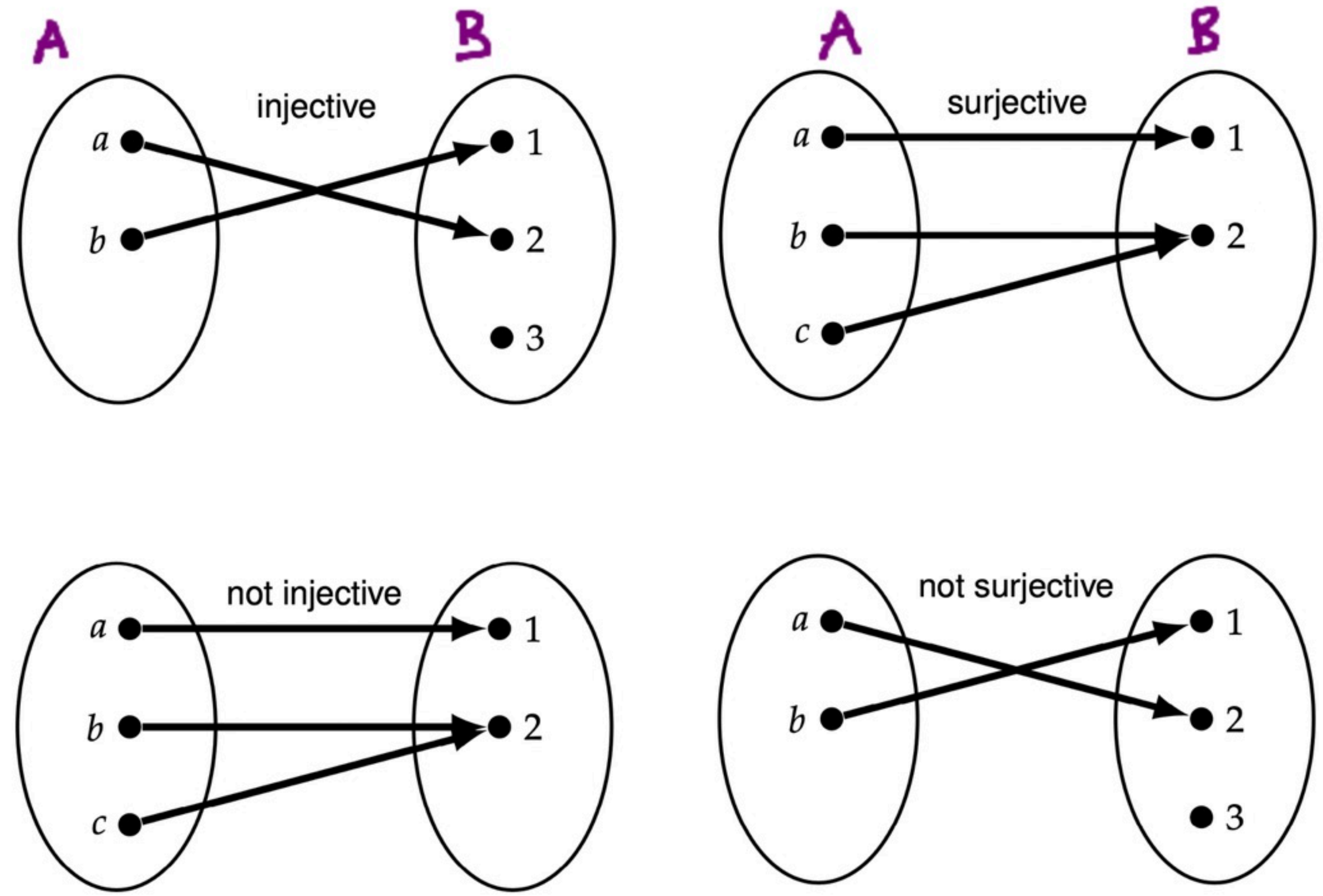
- A: January
- B: February
- C: March
- D: April
- E: May
- 👍 June
- ✋ July
- ❤️ August
- 😊 September
- 🦤 October
- 🐢 November
- 🐍 December

16

Rule #1 (Mapping Rule): relating the cardinalities of two sets.

$|A| \leq |B|$

Injective: *one-to-one* **Surjective:** *onto*



$|A| \geq |B|$

bijective:

injective + surjective

$|A| \leq |B| \wedge |A| \geq |B|$

$\rightarrow |A| = |B|$

main idea: count the items in a set we know how to count to infer the number of items in another set.

Main idea: count something we know, infer the size of the other.

When preparing for Halloween, you bought 4 bags of candies with 20 pieces of candy in each bag. Assume you handed out exactly one candy to each trick-or-treater. At the end of the night, there were three candies left. How many trick-or-treaters were there?

total candies = $4 \times 20 - 3 = 77$

Rule #2: Product rule.

- Event A can occur in m ways.
- Event B can occur in n ways.
- Total number of possible events in which $a \in A \wedge b \in B$ happen: $m \times$n

	a_1	a_2	a_3	...	a_m
b_1	(a_1, b_1)	(a_2, b_1)	...	...	(a_m, b_1)
b_2	$\vdots$				
$\vdots$					
b_n	(a_1, b_n)	---	---	---	(a_m, b_n)

Example: Number of possible Vermont license plates?
(3 letters and then 3 numbers)

a-z a-z a-z 0-9 0-9 0-9

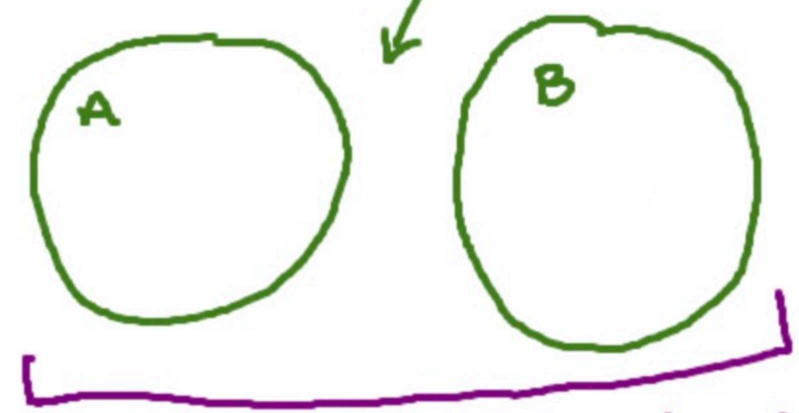
$$26 \times 26 \times 26 \times 10 \times 10 \times 10 = 26^3 10^3$$

$$= 17{,}576{,}000$$

Rule #3: Addition rule.

- Event A can occur in m ways.
- Event B can occur in n ways.
- Assume A and B are **disjoint**.
- A **or** B occurs in $m + n$ ways.

no overlap (disjoint)



$$|A \cup B| = |A| + |B|$$

$$m + n$$

Example: Number of possible passwords? Assume passwords must:

- contain 6-8 characters,
- each character is either an uppercase letter or a number,
- must contain at least one number.

let P_n be # passwords of length n

count $P_6 + P_7 + P_8$

_ _ _ _ _ _ → 36^n possibilities

options?

A-Z, 0-9

$\underbrace{\text{A-Z, 0-9}}_{26+10=36 \text{ options}}$

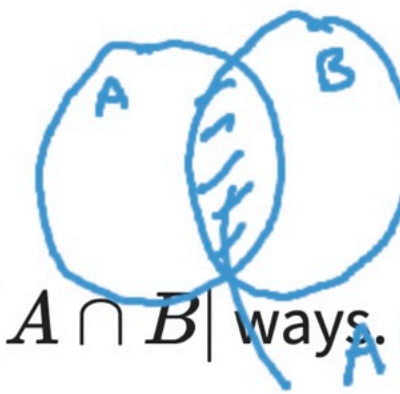
subtract # passwords with <u>no</u> numbers : 26^n

$P_n = 36^n - 26^n$

$\# \text{passwords} = (36^8 - 26^8) + (36^7 - 26^7) + (36^6 - 26^6)$

Rule #4: Principle of Inclusion-Exclusion (PIE).

- Event A can occur in $|A|$ ways.
- Event B can occur in $|B|$ ways.
- Assume A and B are **not disjoint**.
- A **or** B occurs in $|A \cup B| = |A| + |B| - |A \cap B|$ ways.



$A \cap B$

$|A \cup B| = |A| + |B| - |A \cap B|$

Example: How many 5-bit strings $\underbrace{\text{start with 1}}_{A}$ or $\underbrace{\text{end with 00}}_{B}$?

00101

$|A|$: 1 x x x x $\rightarrow 2^4 = 16$

$|B|$: x x x 0 0 $\rightarrow 2^3 = 8$

$|A \cap B|$: 1 x x 0 0 $\rightarrow 2^2 = 4$

PIE: $16 + 8 - 4 = 20$ bit strings

The Pigeonhole Principle:

In general: If n objects are placed into k boxes, then there is at least one box which contains $\geq \lceil \frac{n}{k} \rceil$ objects.

$n = 16$
$k = 12$

Example: Birth month question from beginning of class.

$\left\lceil \frac{16}{12} \right\rceil = 2$

Exercise 1: counting bit strings.

Let $T(n)$ be the number of bit strings of length n that **DO NOT** contain two consecutive zeros.

0
1
2

00 x
10
01
11
3

- Base cases: $T(1) = 2, T(2) = 3.$
- Determine a linear recurrence relation for $T(n)$ in terms of $T(n-1), T(n-2)$.

Let b_n be a bit string of length n without consecutive zeros. Each b_n either ends in 0 or 1.

Note: we can create b_n from b_{n-1}, appending either 0 or 1 (without creating consecutive zeros).

$\overbrace{\underbrace{\underbrace{\_\ \_\ \_\ \_\ \_\ \_}_{b_{n-2}}\ \_}_{b_{n-1}}\ \_}^{b_n}$

case 1: b_n ends with 1 : $b_{n-1}\{\underline{1}\}$ → $T(n-1)$ of these (1 at end doesn't create consecutive 0's)

bit strings without consecutive zeros $= T(n-1)$

case 2: b_n ends with 0 : second to last bit must end with 1

why? consider $b_{n-2}\{\_\ \underline{0}\}$ → all options are

00 x has consecutive zeros
10 ✓ $T(n-2)$ of these
01
11 included in $T(n-1)$ (case 1)

$\boxed{T(n) = T(n-1) + T(n-2)}$

Exercise 2: counting more bit strings.

Let $T(n)$ be the number of bit strings of length n that have two consecutive zeros. Consider a recurrence relation for $T(n)$.

- Base cases: $T(1) = 0, T(2) = 1$.
- Determine a linear recurrence relation for $T(n)$ in terms of $T(n-1), T(n-2)$ and possibly n.
- Use the recurrence relation to calculate $T(5)$.

Let b_n be a bit string of length n with two consecutive zeros. Each b_n either ends in 0 or 1.