CSE 312

Foundations of Computing II

Lecture 1: Introduction & Counting

https://cs.washington.edu/312

Instructors

Stefano Tessaro [he/him]

tessaro@cs

Specialty: **Cryptography**

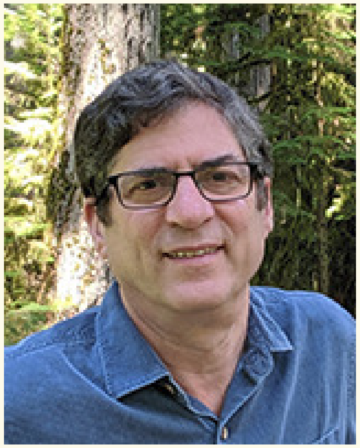
https://homes.cs.washington.edu/~tessaro

Paul Beame [he/him]

beame@cs

Specialty: **Complexity**

https://homes.cs.washington.edu/~beame

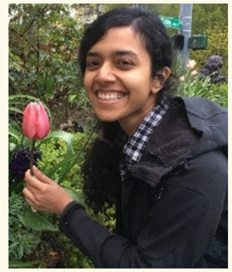


A Team of fantastic TAs

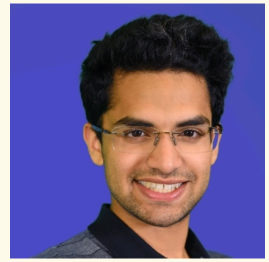
Jan Buzek

Shreya Jayaraman

Aleks Jovcic



Swati Padmanabhan



Jerome Paliakkara

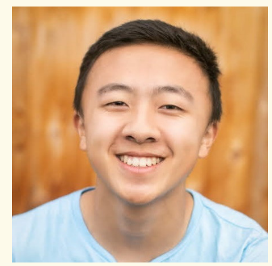
Francis Peng

Phawin Prongpaophan

Tanmay Shah

Chloe Winston

Claris Winston



Ben Zhang

See https://cs.washington.edu/312/staff.html to learn more about their backgrounds and interests!

Masking / COVID-19 policy

We follow the UW's Face Covering Policy

- Masking is not required, it is your choice!
  - Exception: Masking is required for a period of at least 10 days for anyone with recent COVID-19 or recent COVID-19 exposure.
- Masking is still recommended, in particular in a crowded setting
- Masking is strongly recommended for the first two weeks of the quarter

Lectures and Sections

- **Lectures MWF**
  - 9:30-10:20am (Stefano) or 1:30-2:20pm (Paul)
  - Classes will be in person
  - Panopto recording and video released after class – also, live streaming!
  - Annotated slides also uploaded.
- **Poll Everywhere**
  - We will sometimes use Poll Everywhere during class
  - As of this quarter it requires that you sign up directly
- **Sections Thu (starts this week)**
  - Not recorded
  - Will prepare you for problem sets!

Go to https://www.polleverywhere.com/login and login using YOURNETID@uw.edu

Questions and Discussions

- **Office hours throughout the week (starting this Friday)**
  - See https://cs.washington.edu/312/staff.html
- **Ed Discussion**
  - You should have received an invitation (synchronized with the class roaster)
  - Material (resources tab)
  - Announcements (discussion tab)
  - Discussion (discussion tab)

Use Ed discussion forum as much as possible. You can make private posts that only the staff can view! Email instructors for personal issues.

Engagement

- **"Concept checks" after each lecture   5-8 %**
  – Must be done (on Gradescope) before the next lecture by 9:00am
  – Simple questions to reinforce concepts taught in each class
  – Keep you engaged throughout the week, so that homework becomes less of a hurdle
- **8 Problem Sets (Gradescope)   45-50 %**
  – Solved individually.   Discussion with others allowed but separate solutions
  – Generally due Wednesdays starting next week
  – First problem set posted this week before section
- **Midterm 15-20 %**
  – In class on **Wednesday, May 4**
- **Final Exam 30-35 %**
  – **Monday, June 6** either **2:30-4:20** or **4:30-6:20** location **TBA**
  – **For the A section this is a different time from the one on the UW final exam timetable**

Check out the syllabus for policies on late submission for checkpoints and HW

More details see

Course Webpage **https://cs.washington.edu/312**

Foundations of Computing II

=

Introduction to Counting, Probability & Statistics

for computer scientists

What is probability??
Why probability?!

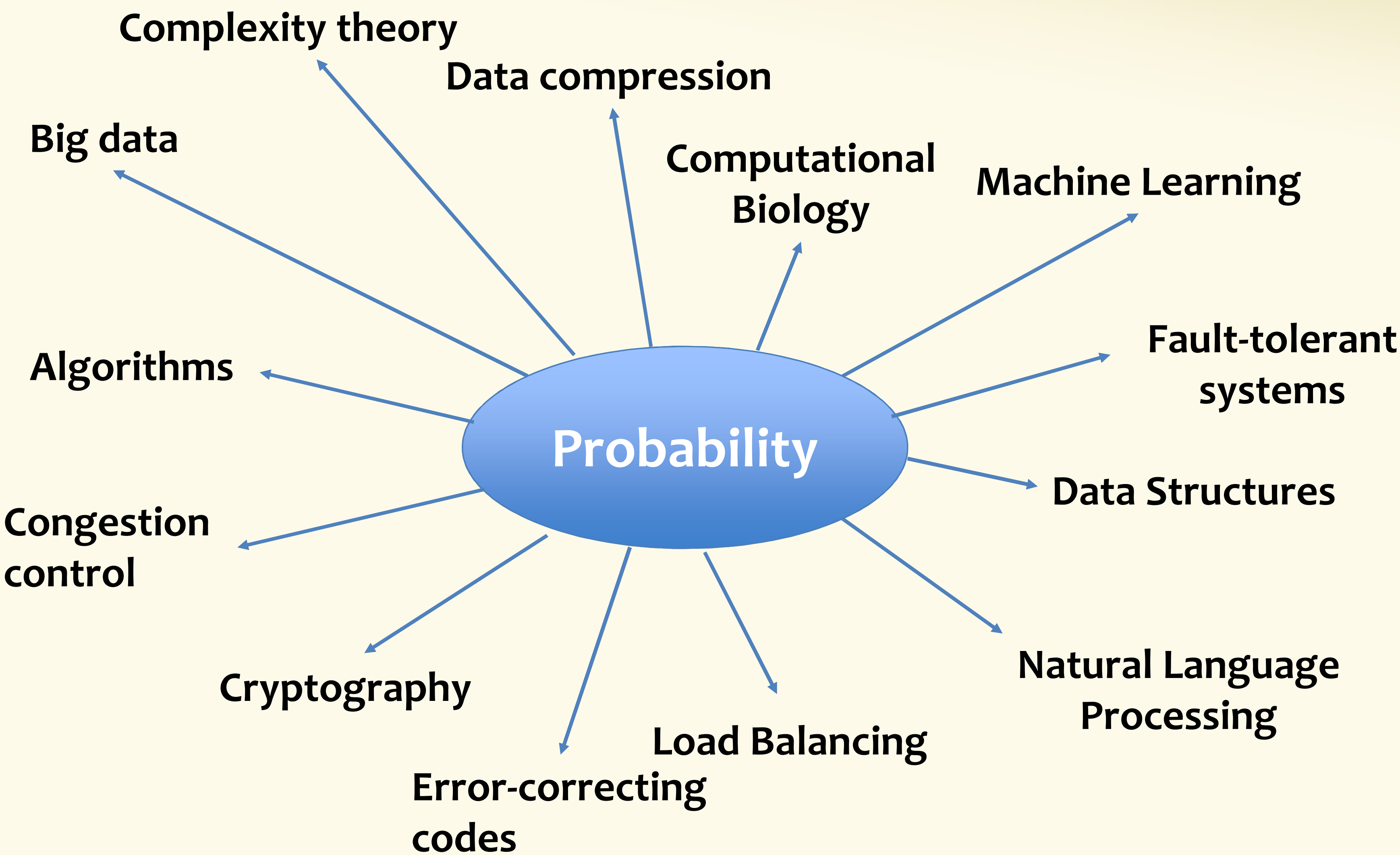
Probability is our tool for understanding random processes

- Randomness in nature and the sciences/social sciences
  - At the quantum level, everything is random
  - Best way to understand and simulate behavior of complex systems
  - A way to design and understand experiments, observations
    - In the lab, the field, medical trials, surveys
- In Computer Science, randomness has these kinds of roles but also important new ones…

Probability and randomness in Computer Science

- Understanding/modelling the inputs to and behavior of our algorithms
  - In ML, program testing/fuzzing, algorithm analysis, …
- Experiments to validate our designs
  - In user studies, HCI, CS applications in other fields, …
- A tool for hiding information, protecting against adversaries/failures
  - Cryptography, privacy, fault tolerance, computer security, …
- A tool for simpler and more efficient design
  - Data structures, algorithms, ML, …
- …

+ much more!


Complexity theory
Data compression
Big data
Computational Biology
Machine Learning
Algorithms
Fault-tolerant systems
Probability
Data Structures
Congestion control
Cryptography
Natural Language Processing
Load Balancing
Error-correcting codes

Content

- Counting (basis of discrete probability)
  - Counting, Permutation, Combination, inclusion-exclusion, Pigeonhole Principle
- What is probability
  - Probability space, events, basic properties of probabilities, conditional probability, independence, expectation, variance
- Properties of probability
  - Various inequalities, Zoo of discrete random variables, Concentration, Tail bounds
- Continuous Probability
  - Probability Density Functions, Cumulative Density Functions, Uniform, Exponential, Normal distributions, Central Limit Theorem, Estimation
- Applications
  - A sample of randomized algorithms, differential privacy, learning …

Today: Counting

We are interested in counting the number of objects with a certain given property.

"How many ways are there to assign 7 TAs to 5 sections, such that each section is assigned to two TAs, and no TA is assigned to more than two sections?"

"How many positive integer solutions (x, y, z) *does the equation* $x^3 + y^3 = z^3$ *have?"*

Generally: Question boils down to computing <u>cardinality</u> $|S|$ of some given set S.

(Discrete) Probability and Counting are Twin Brothers

"What is the probability that a random student from CSE312 has black hair?"

$$= \frac{\#\ students\ with\ black\ hair}{\#students}$$

Today – Two basic rules

- Sum rule
- Product rule

Sum Rule

If you can choose from

- EITHER one of n options,
- OR one of m options with NO overlap with the previous n

then the number of possible outcomes of the experiment is

$$n + m$$

Counting "lunches"

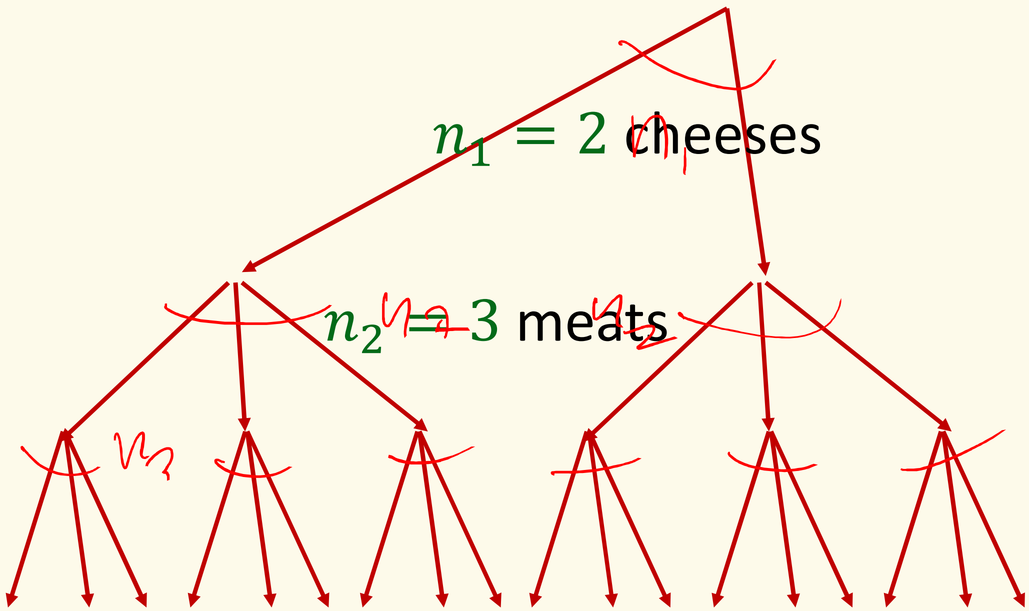
If a lunch order consists of **either** one of 6 soups **or** one of 9 salads, how many different lunch orders are possible?

Product Rule: In a sequential process, there are

- n_1 choices for the first step,
- n_2 choices for the second step (given the first choice), …, and
- n_m choices for the m^{th} step (given the previous choices),

then the total number of outcomes is $n_1 \times n_2 \times \cdots \times n_m$

Example: "How many subs?"



$n_3 = 3$ veggies

$2 \times 3 \times 3 = 18$

Product rule example – Laptop customization

Alice wants to buy a new laptop.

- The laptop can be **blue**, **orange**, **purple**, or **silver**.
- The SSD storage can be **128GB**, **256GB**, and **512GB**
- The available RAM can be **8GB** or **16GB**.
- The laptop comes with a **13”** or with a **15”** screen.

How many different laptop configurations are there?

$$\boxed{4} \times \boxed{3} \times \boxed{2} \times \boxed{2} = \boxed{48}$$

Product rule example – Strings

How many string of length 5 over the alphabet $\{A, B, C, \ldots, Z\}$ are there?

- E.g., AZURE, BINGO, TANGO, STEVE, SARAH, …

$26 \times 26 \times 26 \times 26 \times 26 = 26^5$

How many binary string of length n over the alphabet $\{0,1\}$?

- E.g., $0 \cdots 0, 1 \cdots 1, 0 \cdots 01, \ldots$

$2 \times 2 \times 2 \times \cdots \times 2 = 2^n$

Product rule example – Cartesian Product

Definition. The **cartesian product** of two sets S, T is

$$S \times T = \{(a, b): a \in S, b \in T\}$$

Called a 2-sequence
Order matters! $(a, b) \neq (b, a)$

(a, b)

$|S \times T| = |S| \times |T|$

$|A_1 \times A_2 \times \cdots \times A_n| = |A_1| \times |A_2| \times |A_3| \times \cdots \times |A_n|$

Product rule example – Power set

Definition. The **power set** of S is the set of all subsets of S,

$$\{X : X \subseteq S\}.$$

Notations: $\mathcal{P}(S)$ or simply 2^S (which we will use).

Example.
$2^{\{\bigstar, \spadesuit\}} = \{\emptyset, \{\bigstar\}, \{\spadesuit\}, \{\bigstar, \spadesuit\}\}$
$2^{\emptyset} = \{\emptyset\}$
…

How many different subsets of S are there if $|S| = n$?

Product rule example – Power set

set $S = \{e_1, e_2, e_3, \cdots, e_n\}$

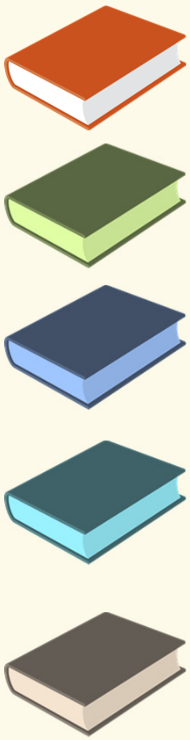
subset $X = \{$ e_1 or ~~e_1~~, e_2 or ~~e_2~~, e_3 or ~~e_3~~ ... e_n or ~~e_n~~ $\}$

$\boxed{2} \times \boxed{2} \times \boxed{2} \times \cdots \times \boxed{2} = \boxed{2^n}$

Proposition. $|2^S| = 2^{|S|}$

Product rule – One more example

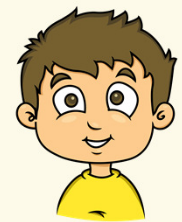
5 books



> ***"How many ways are there to distribute 5 books among Alice, Bob, and Charlie?"***

Every book to one person, everyone gets ≥ 0 books.

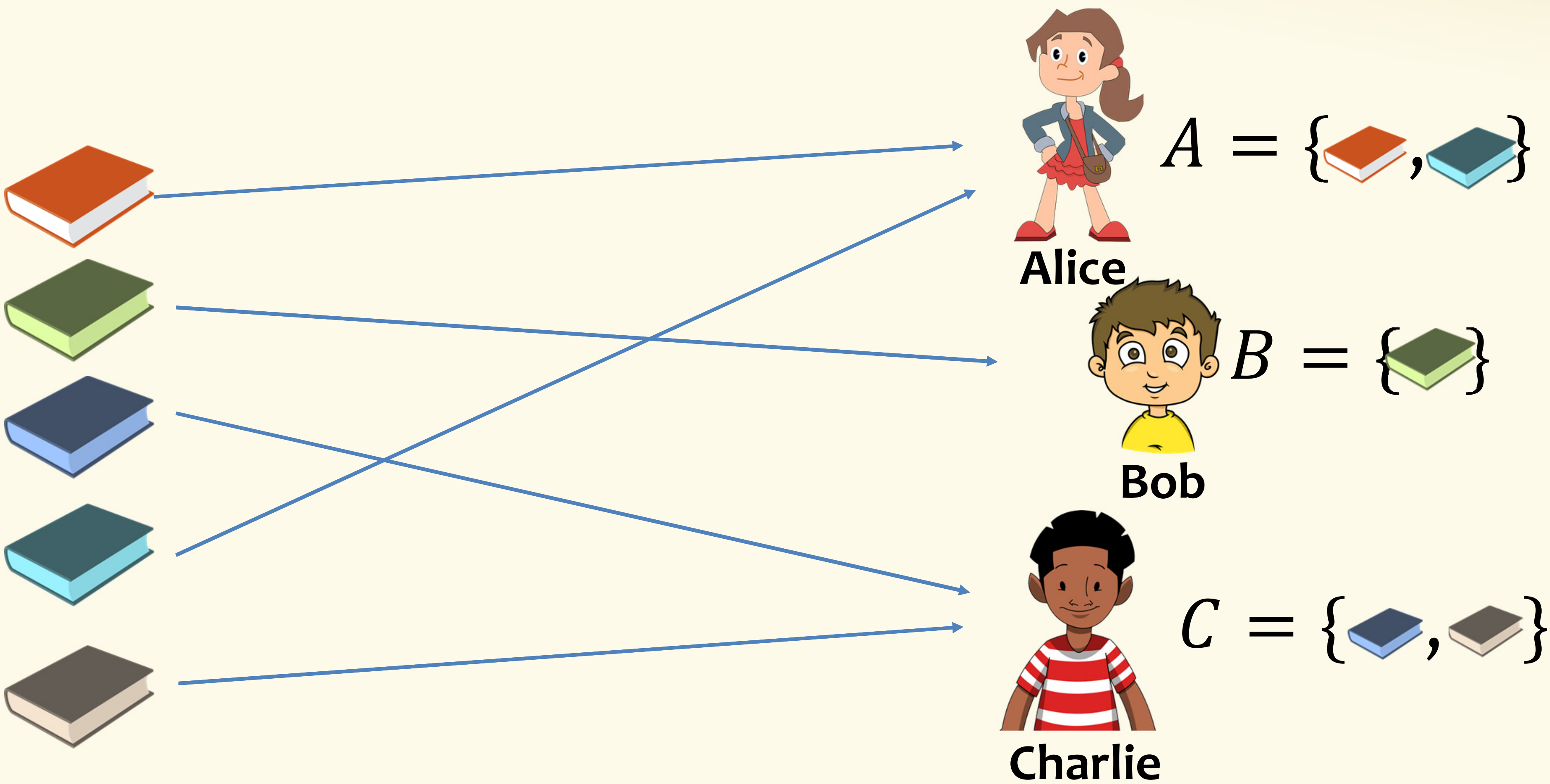
Alice



Bob

Charlie

Example Book Assignment



Book assignment – Modeling

Correct?

Poll:

A. Correct
B. Overcount
C. Undercount
D. No idea

pollev.com/paulbeame028

$2^5 = 32$ options

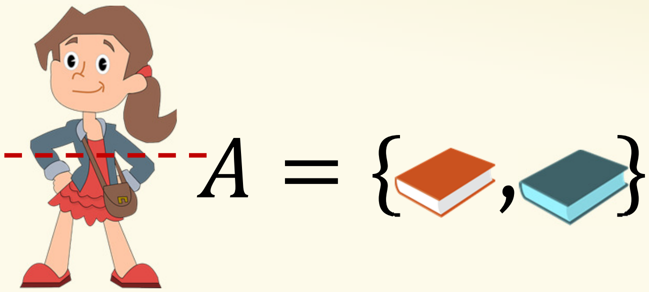
$\times$

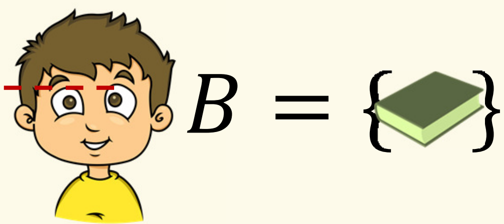
$2^5 = 32$ options

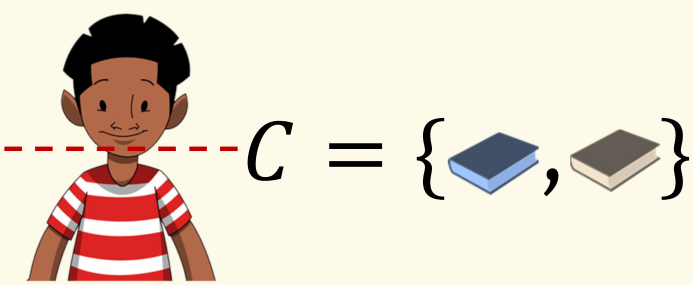
$\times$

$2^5 = 32$ options

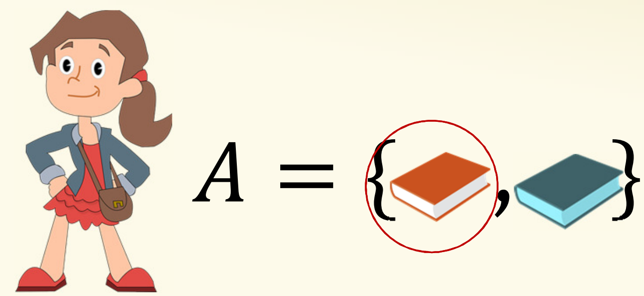
$= 32^3$ assignments



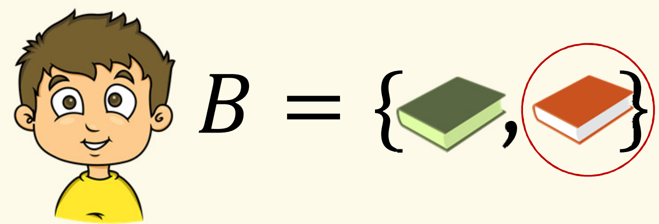




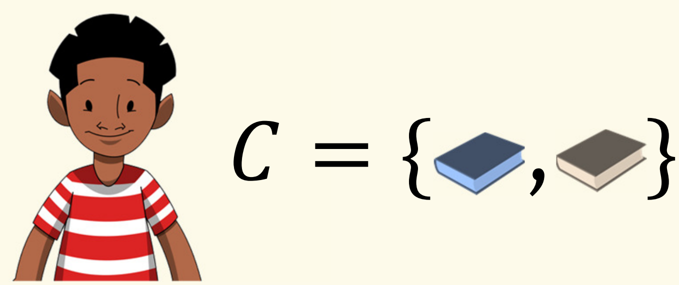
Problem – Overcounting



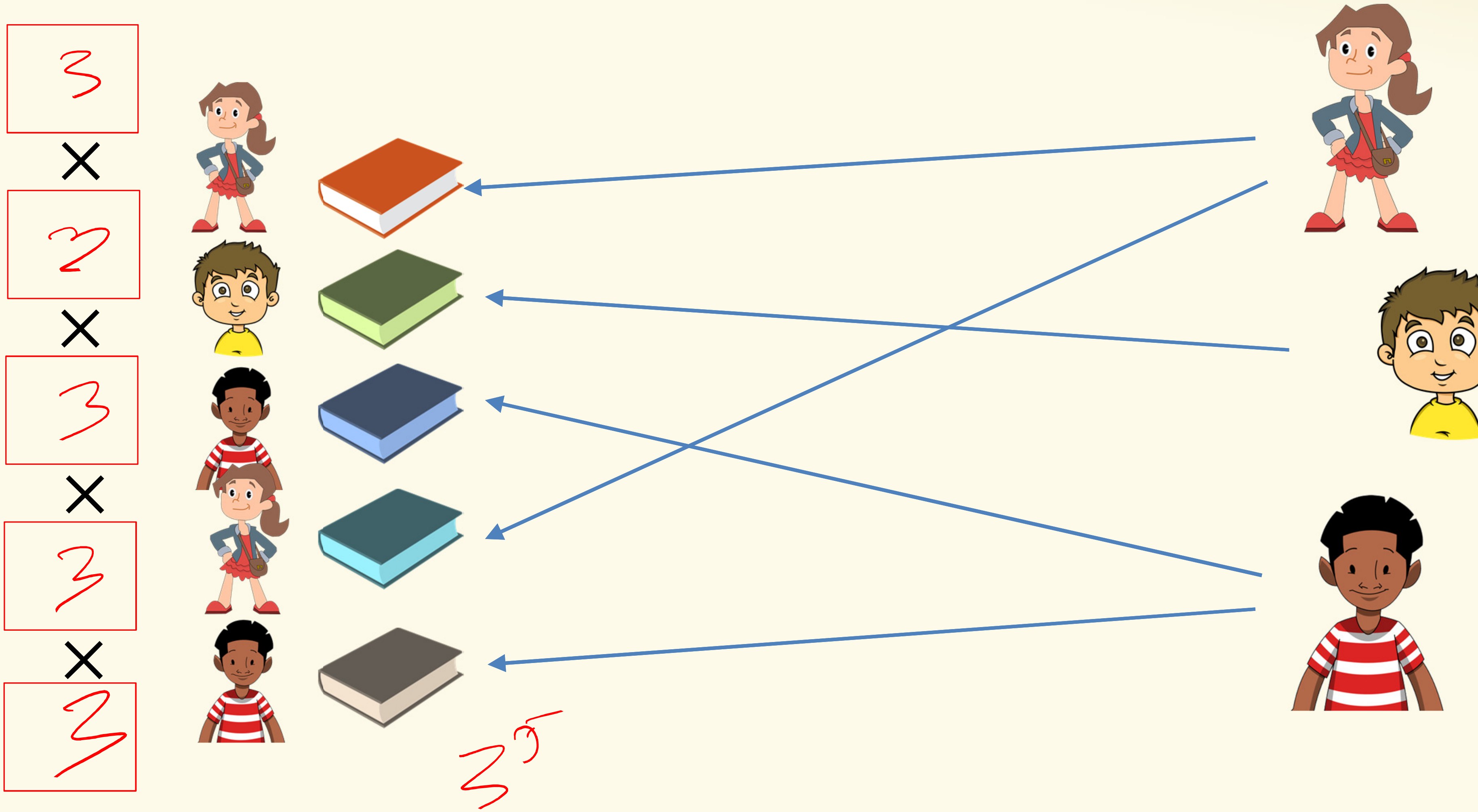
Problem: We are counting some invalid assignments!!!
→ **overcounting**!



What went wrong in the sequential process?
After assigning A to Alice,
B is no longer a valid option for Bob



Book assignments – A Clever Approach



Lesson: Representation of what we are counting is very important!

Tip: Use different methods to double check yourself
Think about counter examples to your own solution.

The first concept check is out and due 9:00am before the next lecture

The concept checks are meant to help you immediately reinforce what is learned.

Students from previous quarters have found them really useful!