CSE 312

Foundations of Computing II

Anna R. Karlin

Welcome!

CSE 312 website – cs.uw.edu/312

Staff

- **Instructor: Anna R Karlin**
  - Professor in the Allen School for n years.
  - Email: karlin@cs.washington.edu
  - Office: CSE1 (Allen) 586
- **Tas:**

Siddharth Baasri	Ronald Lin
Escher Crawford	Arnav Mazumder
Darin Ershov	Medha Mittal
Yao Ching Hsieh	Mayee Sun
Zhiyuan Jia	Jolie Zhou (head TA)
Krisha Khandelwal	

Lectures

- **Lectures MWF**
  - 9:30- 10:20 am
  - Recorded on Panopto
- **Ask questions during class**
  - I love it when the class is interactive! And trust me, you'll like it more too.
  - If you don't understand something, ask!
  - In addition, ask questions in office hours, on edstem, and on our 312 GPT (!!)
  - Help each other out! Try to answer your fellow classmate's questions on edstem!

Questions and Discussions

- **Office hours throughout the week (starting this Friday). Find times on website.**

- **Ed Discussion**
  - You should already have access (synchronized daily with the class roster)
  - Use Ed discussion forum as much as possible. You can make private posts that only the staff can view! Email instructors for personal issues.

Sections

- **Sections on Thursdays**
  - Not recorded, for privacy of student discussion and to motivate you to ask questions.
  - Important way to reinforce concepts and to prepare for doing problem sets.
  - Bring **laptop/phone** and pencil or pen to section. (We will bring paper for you to write on.)

Concept checks

- **Concept checks after each lecture**
  - Released by 11am (30 mins after lecture). Must be done before the next lecture.
  - Simple questions to reinforce concepts taught in each class.
  - Keep you engaged and on top of the material throughout the week, so that homework becomes less of a hurdle.
  - **Do them as soon as you can after lecture. Really important.**

- **Important!!!!**
  - If you don't submit by the due date, you won't be able to see the solutions until at least a week later. **So submit, even if you don't answer any of the questions!! You can resubmit as many times as you want before the deadline.**

Grading etc.

- **Concept checks after each lecture 15%**
- **8 Homeworks (submission on Gradescope) 35%**
  - Mostly math. A few will have coding component.
- **Midterm (20%): Thursday, May 18, 6-7:30pm**
- **Final (30%): Wednesday, June 10, 8:30 - 10:20 am**
- **Small extra credit opportunity:** help your peers out by answering their questions on Ed!

Syllabus

Check out the syllabus for policies on late submission, grading breakdown, academic integrity, office hours, course tools, etc.

Bring questions about syllabus to class on Wednesday (or, better yet, ask on Ed).

Problem Sets

- Part 1 of Pset1 will be out today!!
- The rest of Pset1 will be out on Wednesday.
- The whole thing is due 11:59pm the following Wednesday (April 8).
- In general, every problem set is due one week after it's released.

What is this class about?

Foundations of Computing II

=

Introduction to Probability & Statistics

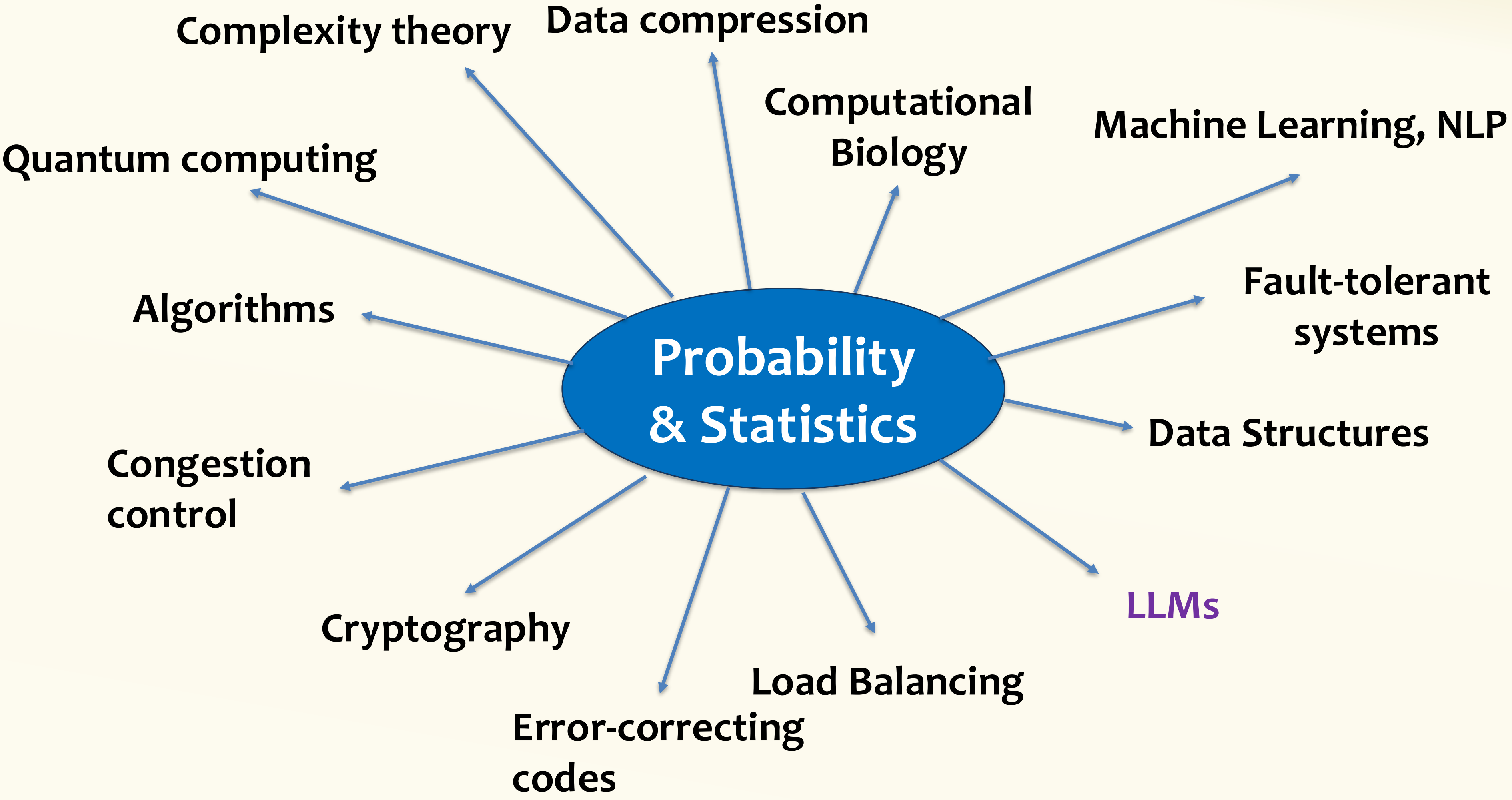
for <u>computer scientists</u>

<u>What</u> is probability??

<u>Why</u> probability?!

Connections

+ much more!



Beyond important applications in computer science and engineering

Two additional arguments for studying hard and really learning the material in this class:

- The economic argument (the “need”)
- The “fun/useful in real life” argument (the “want”)

The economic argument and the elephant in the room

Generative AI (like ChatGPT, Claude or Gemini) can solve 95% of the homework problems in this class in seconds.

Thus, the market value of “getting the answer” is effectively $0.00

Employers do not pay humans for skills that are free via an API

So why are you here?

The economic argument and the elephant in the room

Generative AI (like ChatGPT, Claude or Gemini) can solve 95% of the homework problems in this class in seconds.

Thus, the market value of "getting the answer" is effectively $0.00

Employers do not pay humans for skills that are free via an API

So why are you here?

You are here to learn how to be the **Pilot**, not the **Passenger**

The Pilot vs Autopilot Analogy

Flying a plane on autopilot is easy. Pilots are paid high salaries not for the 99% of the time the weather is clear, but for the 1% of the time the system fails.

- **The gap**

  AI is excellent at the “average” case. It fails catastrophically at “edge cases” – tail risks and black swan events.

- **The skill**

  Probability is your instrument panel. If you don’t know how to read the instruments (e.g., *variance, conditional probability, Bayes theorem*), you cannot take over when the AI enters turbulence.

The "Zero-Value" Proposition

If you use AI to bypass the struggle of learning probability, you are training yourself for a job that no longer exists.

- **The pivot**

  In the future, it is unclear to what extent you will need to generate code/math from scratch. But you absolutely will need to **verify** code and math. And that requires serious training.

- **The reality**

  In your first job, you will likely supervise AI agents. If you can not spot a subtle probabilistic error in their output (e.g., a hallucinated *confidence interval*), you are a liability, not an asset.

The Changing Interview Landscape

Tech interviews are adapting to the post-AI world.

- **The test**

  An interviewer might not ask you to solve a problem. They might ask you to defend a solution.

  Example: “*Why did you model this as a Poisson distribution? What happens to your risk model if the correlation shifts by 0.1?*”

- **The result**

  An LLM can give you the formula, but it cannot give you the intuition to defend it under pressure.

Debugging AI Bugs

Traditional software bugs are logical (True/False). AI bugs are probabilistic (maybe/sometimes)

- **The challenge**

  Debugging a deterministic loop is not necessarily that hard. Debugging a stochastic system that only fails 0.4% of the time requires deep statistical intuition.

- **The consequence**

  If you don’t learn the 312 material, you will struggle to debug the systems of the future.

The “Human in the Loop” Premium

We are handing over critical decisions (loans, medical diagnoses, driving) to probabilistic algorithms.

- **The future**

  The highest paid people will be the **Risk Managers.** These are the people who can mathematically quantify the risk of an algorithm – who can put a *confidence interval* on an AI’s hallucination.

For all of the above reasons, you are NOT allowed to use the usual AI tools in this class (ChatGPT, Gemini, Claude, etc)

BUT…..

We are experimenting with a 312 GPT!!

This course provides an AI learning assistant that works like a **teaching assistant available 24/7.**

It can

- help explain the material we are covering
- walk through section worksheet solutions
- generate extra practice problems
- help you get started if you are stuck on homework.

Like a human TA, the assistant is designed to **guide your reasoning rather than provide solutions** to graded homework problems.

Ps Don't forget that AI tools make mistakes.

The economic argument and the elephant in the room

Generative AI (like ChatGPT, Claude or Gemini) can solve 95% of the homework problems in this class in seconds.

Thus, the market value of "getting the answer" is effectively $0.00

Employers do not pay humans for skills that are free via an API

So why are you here?

You are here to learn how to be the **Pilot**, not the **Passenger**

Nonetheless, to pass this class, you must sign off on the following on Pset 1

I have read and understood the syllabus. In addition, I understand that we are entering an era where AI tools are ubiquitous and powerful. I acknowledge that using these tools to bypass the learning process is an act of self-sabotage.

1. **If I use AI, I will use only the 312 Learning Assistant, and I will use it as a Tutor only, not as a Proxy.** I may use it to explain concepts, generate practice problems, help me get started on homework, while keeping in mind that AI tools make errors. I will not use any AI to generate solutions. I understand that doing so constitutes academic misconduct.

2. **I accept the "Audit" Standard:** I understand that at any time, I may be asked to orally explain and defend a solution I submitted. If I cannot explain why a step was taken, I accept that I have not truly "solved" the problem and that I will get 0 points for that problem.

3. **I am building my "mental model":** I recognize that my future value as a computer scientist depends, among other things, on my ability to detect errors in automated systems. I commit to doing the "heavy lifting" of learning and understanding the math now so that I possess the intuition and knowledge required to audit and debug the AI systems of tomorrow.

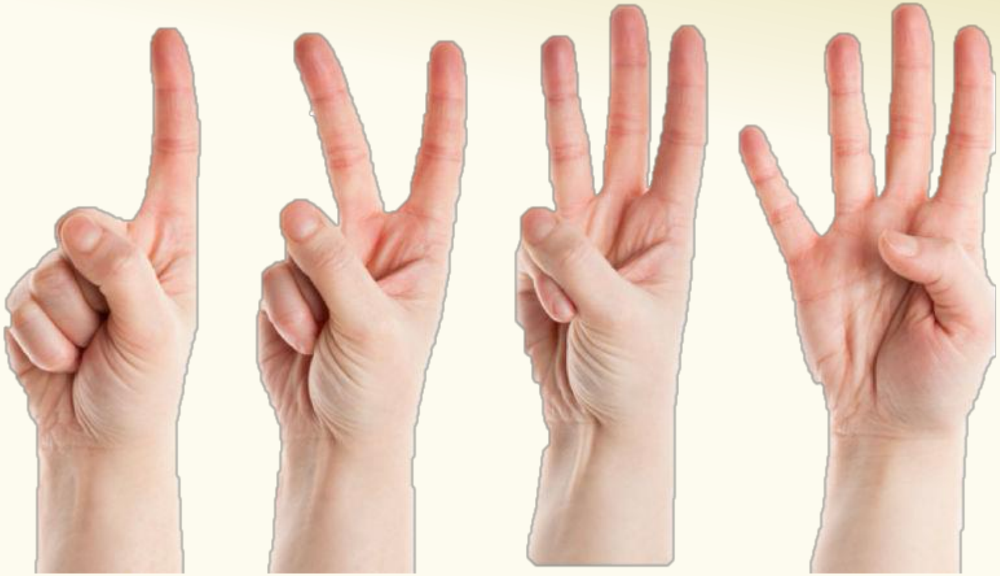
Summary: why you should take this course seriously

- Important applications in computer science and engineering.
- Crucial for success in future CSE (and other) courses.
- Crucial for success in your future career, aka the economic argument (the “need”)
- The “fun/useful in real life” argument (the “want”)

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 & Statistics
  - Probability Density Functions, Cumulative Density Functions, Uniform, Exponential, Normal distributions, Central Limit Theorem, Estimation
- Applications
  - A sample of randomized algorithms, machine learning, differential privacy, …

Counting



CSE 312 Spring 26

Lecture 1

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 integer solutions $(x, y, z) \in \mathbb{Z}^3$ does the equation $x^3 + y^3 = z^3$ have?"

Generally: Question boils down to computing cardinality $|S|$ 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 is

$$n + m$$

Counting "lunches"

If your starter can be **either** one soup (6 choices) **or** one salad (8 choices), how many possible starters?

$6 + 8 = 14$

The Product Rule

Product Rule: If each outcome is constructed by a sequential process where there are

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

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

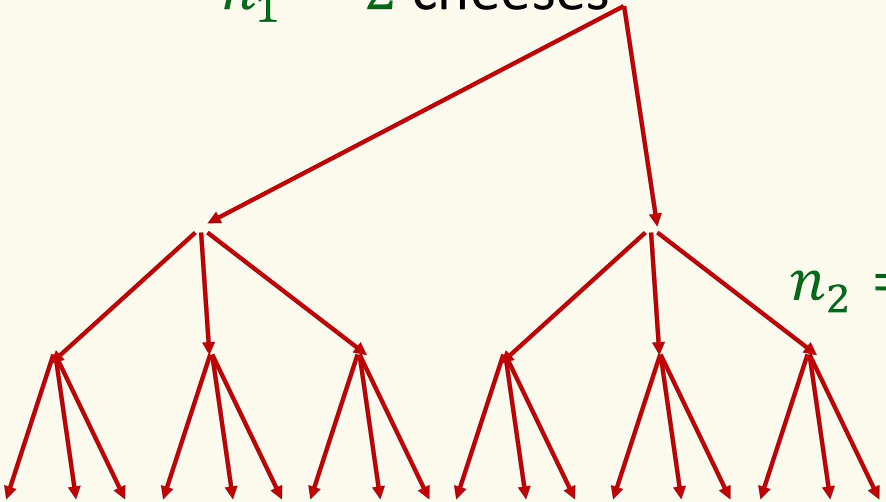
Product Rule (again)

In a sequential process, if there are

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

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

$n_1 = 2$ cheeses



$n_2 = 3$ meats

$n_3 = 3$ veggies

Example: "How many subways?"

$\boxed{2} \times \boxed{3} \times \boxed{3} = \boxed{18}$

Example – Strings

How many strings of length 5 over the alphabet $\underbrace{\{A, B, C, \dots, Z\}}_{26}$ are there?

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

$\boxed{26} \times \boxed{26} \times \boxed{26} \times \boxed{26} \times \boxed{26} = \boxed{26^5}$

Example – Binary Strings

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

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

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

Example – Power set

Definition. The **power set** of S is

$$2^S \stackrel{\text{def}}{=} \{X : X \subseteq S\}$$

Example.

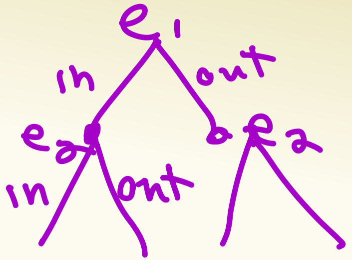
$S = \{★, ♠\}$ $\qquad 2^{\{★,♠\}} = \{\emptyset, \{★\}, \{♠\}, \{★, ♠\}\}$

$S = \emptyset \qquad 2^{\emptyset} = \{\emptyset\}$

…

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

Example – Power set – number of subsets of S



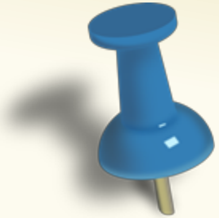
$$S = \{e_1, e_2, e_3, \cdots, e_n\}$$

What is the number of subsets of S, i.e., $|2^S|$?

is e_1 in/out is e_2 in/out is e_3 in/out is e_n in or out

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

Example – ATMs and Pin codes



- How many 4 –digit pin codes are there?
- Each digit one of $\underbrace{\{0, 1, 2, \ldots, 9\}}_{10}$

0011

$\boxed{10} \times \boxed{10} \times \boxed{10} \times \boxed{10} = \boxed{10^4}$

# possible **first** digits	# possible **second** digits	# possible **third** digits	# possible **fourth** digits	# possible **pins**

Example – ATMs and Pin codes – Stronger Pins

- How many **10–digit** pin codes are there with no repeating digit?
- Each digit one of {0, 1, 2,…, 9}; must use each digit ***exactly once***

$\boxed{10} \times \boxed{9} \times \boxed{8} \times \ldots \times \boxed{} \times \boxed{1} = \boxed{10!}$

possible **first** digits

possible **second** digits

possible **third** digits

…

possible **pins**

Permutations

"How many ways to order n distinct objects?"

Answer = $n \times (n-1) \times (n-2) \times \cdots \times 2 \times 1$

Definition. The **factorial function** is

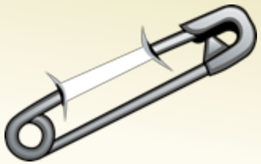
$$n! = n \times (n-1) \times \cdots \times 2 \times 1$$

Note: $0! = 1$

Read as ``n factorial"

Huge: Grows exponentially in n

Example – ATMs and Pin codes – Tricky Pins



- How many 10–digit pin codes with **at least one digit repeated once**?
- Examples: 1111111111, 1234567889, 1353483595

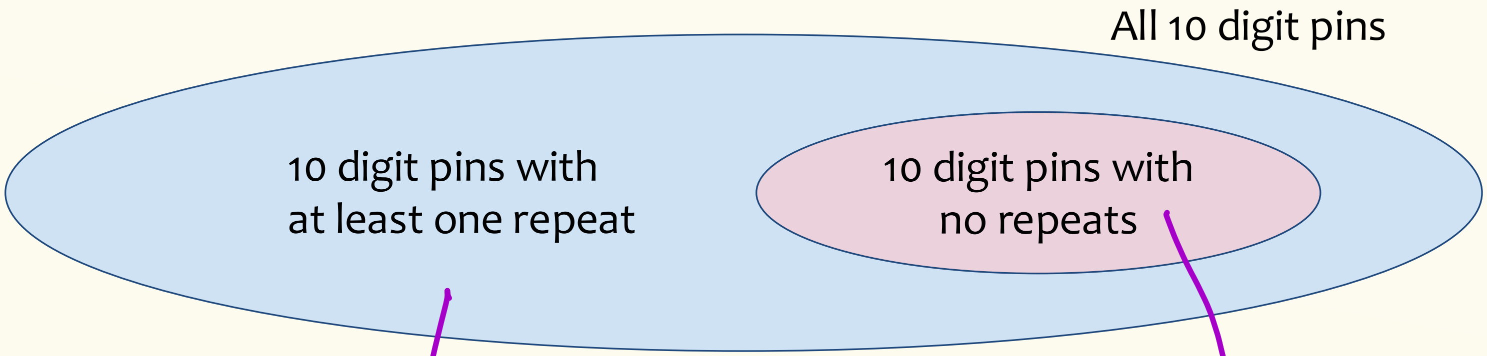
$\square \times \square \times \square \times \ldots \times \square \times \square = \square$

# possible **first** digits	# possible **second** digits	# possible **third** digits	…	# possible **pins**


first digit.
0
1
9
0
1
9

Example – ATMs and Pin codes – Tricky Pins illustrated

- How many 10–digit pin codes with at least one digit repeated once?



10^{10}

$10^{10} - 10!$

$10!$

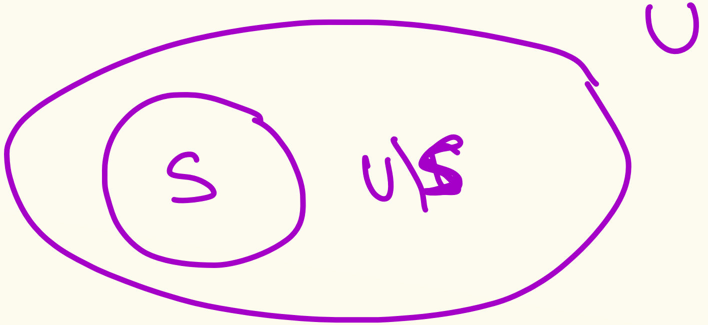
Complementary Counting

Let U be a set and S a subset of interest

Let U \S denote the set difference (the part of U that is not in S)

Then $|U\backslash S| = |U| - |S|$

$|S| = |U| - |U\backslash S|$



Hint that you should consider using complementary counting:
When you see the phrases “at least” or “some”

Distinct Letters

"How many sequences of 5 distinct alphabet letters from $\{\underbrace{A, B, \dots, Z}_{26}\}$*?"*

E.g., AZURE, BINGO, TANGO. But not: STEVE, SARAH

$$\boxed{26} \times \boxed{25} \times \boxed{24} \times \boxed{23} \times \boxed{22} = \frac{26!}{21!}$$

Distinct Letters – answer

"How many sequences of 5 *distinct alphabet letters from* $\{A, B, \dots, Z\}$*?"*

E.g., AZURE, BINGO, TANGO. But not: STEVE, SARAH

Answer: $26 \times 25 \times 24 \times 23 \times 22 =$ 7893600

$\frac{26!}{21!}$

In general

Aka: k-permutations

Fact. # of ways to arrange k out of n distinct objects in a sequence.

$$P(n,k) = \underset{\text{1st}}{n} \times \underset{\text{2nd}}{(n-1)} \times \cdots \times \underset{k\text{th}}{(n-k+1)} = \frac{n!}{(n-k)!}$$

We sometimes say ``n pick k''

Number of Subsets

"How many size-5 ***subsets*** *of* $\{A, B, \ldots, Z\}$*?"*

E.g., {A,Z,U,R,E}, {B,I,N,G,O}, {T,A,N,G,O}. But not: {S,T,E,V}, {S,A,R,H},…

Number of Subsets – Not the same as k-permutations.

*"How many size-5 **subsets** of* $\{A, B, \dots, Z\}$*?"*

E.g., {A,Z,U,R,E}, {B,I,N,G,O}, {T,A,N,G,O}. But not:
{S,T,E,V}, {S,A,R,H},...

Different from k-permutations: **NO ORDER**

Different sequences: TANGO, OGNAT, ATNGO, NATGO, ONATG ...

Same set: {T,A,N,G,O}, {O,G,N,A,T}, {A,T,N,G,O}, {N,A,T,G,O}, {O,N,A,T,G}... ...

How to count number of 5 element subsets of $\{A, B, \ldots, Z\}$?

Consider the following process:

1. Choose an **unordered** subset $S \subseteq \{A, B, \ldots, Z\}$ of size $|S| = 5$. e.g. $S = \{A, G, N, O, T\}$
1. Choose a permutation of letters in S
   e.g., *TANGO, AGNOT, NAGOT, GOTAN, GOATN, NGOAT, …*

Outcome: An **ordered** sequence of 5 distinct letters from $\{A, B, \ldots, Z\}$

$\boxed{???}$

$\times$

$\boxed{5!}$

$=$

$\boxed{\frac{26!}{21!}}$

$$\underbrace{???}_{\#\text{unordered subsets}} \times 5! = \frac{26!}{21!}$$

Number of Subsets – Fill in what we know

Consider the following process:

1. Choose an **unordered** subset $S \subseteq \{A, B, \dots, Z\}$ of size $|S| = 5$. e.g. $S = \{A, G, N, O, T\}$
1. Choose a permutation of letters in S
   e.g., *TANGO, AGNOT, NAGOT, GOTAN, GOATN, NGOAT, …*

$$\boxed{???} \times \boxed{5!} = \boxed{\frac{26!}{21!}}$$

Outcome: An **ordered** sequence of 5 distinct letters from $\{A, B, \dots, Z\}$

$$??? = \frac{26!}{21!\,5!} = 65780$$

Number of Subsets -- don't care about order

Fact. The number of subsets of size k of a set of size n is

$$C(n,k) = \binom{n}{k} = \frac{n!}{k!(n-k)!}$$

we say "n choose k"

[also called **combinations** or **binomial coefficients**]

Quick Summary

- 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 $n + m$

- Product Rule

  In a sequential process, if there are

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

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

- Complementary Counting

Quick Summary, continued

- K-sequences: How many length k sequences over alphabet of size n? repetition allowed.
  - Product rule ➔ n^K
- K-permutations: How many length k sequences over alphabet of size n, without repetition?
  - Permutation ➔ $\frac{n!}{(n-k)!}$
- K-combinations: How many size k subsets of a set of size n (without repetition and without order)?
  - Combination ➔ $\binom{n}{k} = \frac{n!}{k!(n-k)!}$

The first concept check (CC) will be out no later than 11am and is due before 9:30am on Wednesday

The concept checks are meant to help you immediately reinforce what is learned in each lecture. (Today's CC also reviews summation and product notation.)

Students from previous quarters found them really useful!