CSCI 2824: Discrete Structures

Lecture 26: Count-ing

Reminders

Submissions:

- Homework 9: **Fri 11/01 at noon -** Moodle

Readings:

- Ch. 6 – Counting

Midterm II

- Tuesday November 5th

Last time

- Strong Induction
- Recursion

Today:

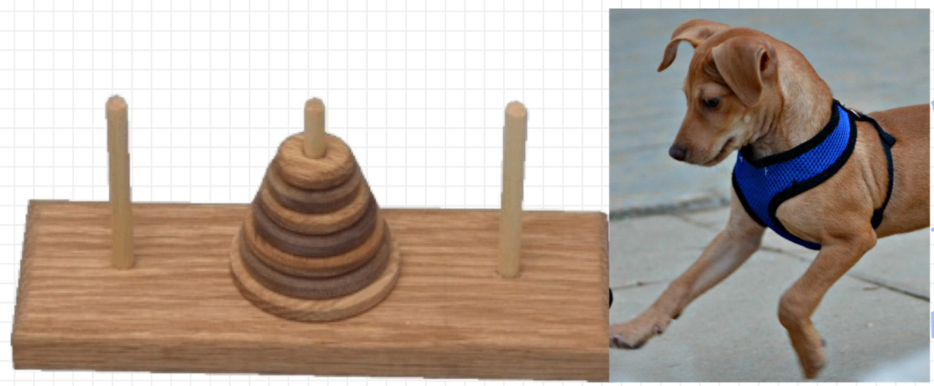
- Recursion, cont.
- Let’s start counting

Recursion

Recursion – Towers of Hanoi

How many times has this happened to you? You're just walking along, minding your own business, when you come across **three** pegs, one of which has a **stack of disks** on it!

Goal: Move the tower from one peg to another



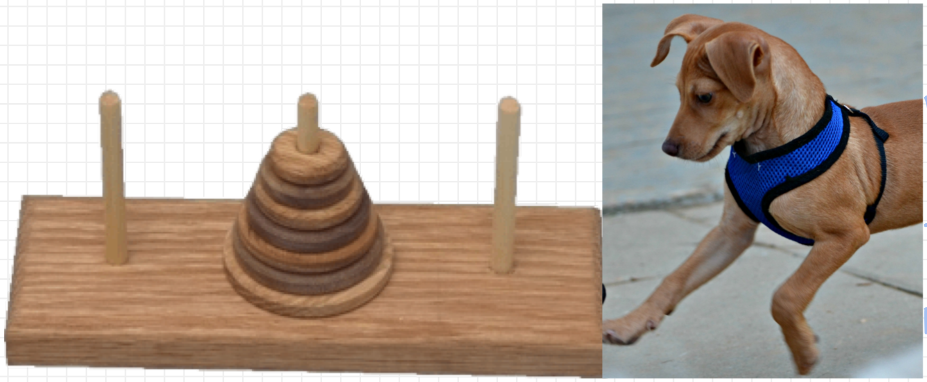
Rules:

- Move one disk at a time
- No larger disk on top of a smaller disk

Recursion – Towers of Hanoi

How many times has this happened to you? You're just walking along, minding your own business, when you come across **three** pegs, one of which has a **stack of disks** on it!

Goal: Move the tower from one peg to another



Rules:

- Move one disk at a time
- No larger disk on top of a smaller disk

Question: $M(n)$ = # distinct moves to move an n-disk Tower = ???

Recursion – Towers of Hanoi

How many times has this happened to you? You're just walking along, minding your own business, when you come across **three** pegs, one of which has a **stack of disks** on it!

Goal: Move the tower from one peg to another

Rules:

- Move one disk at a time
- No larger disk on top of a smaller disk

Question: $M(n)$ = # distinct moves to move an n-disk Tower = ???

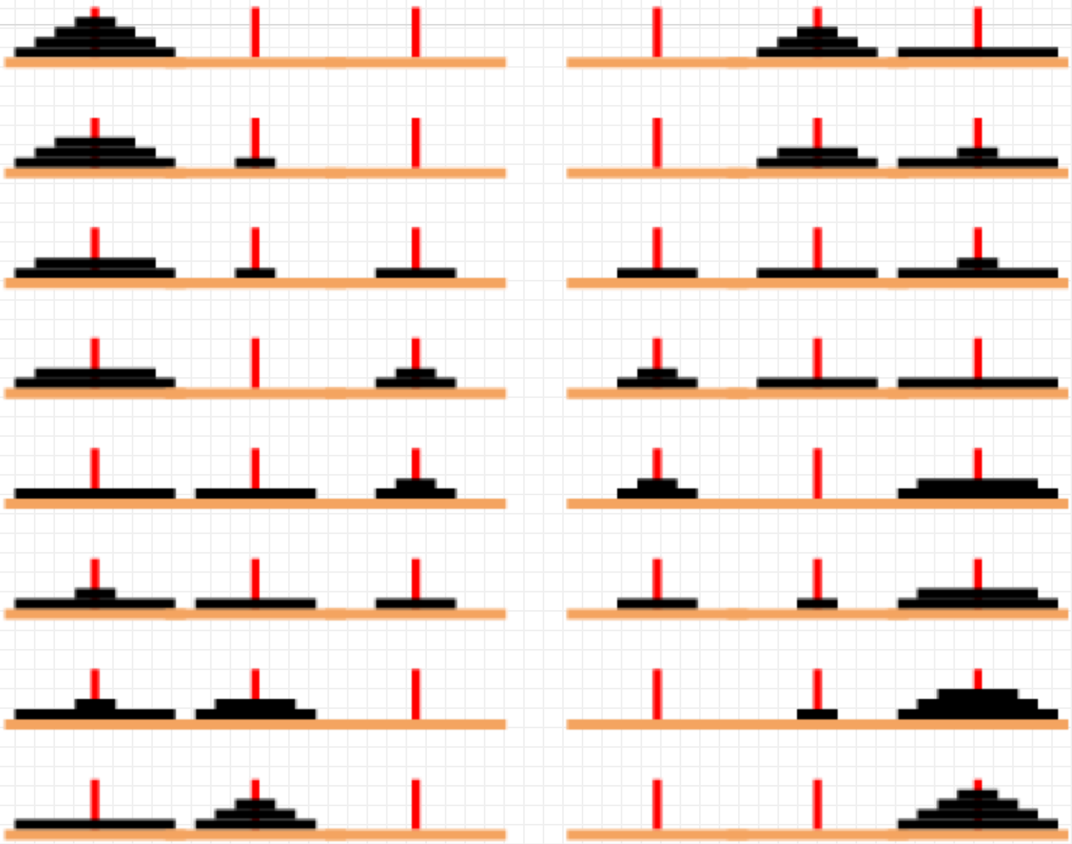
Strategy: 1. Define a recurrence for the number of moves

2. Conjecture a closed-form expression

3. Prove it with induction and the recurrence

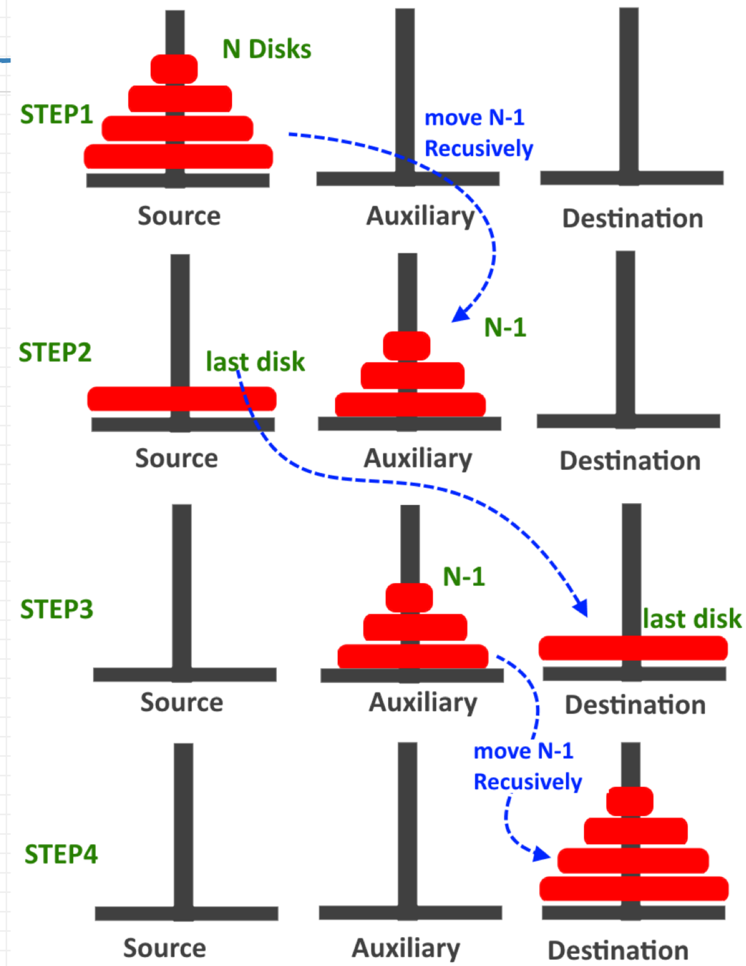
Recursion – Towers of Hanoi

What does a solution even look like?



Recursion – Towers of Hanoi

… and how's this going to work?



Recursion – Towers of Hanoi

Let $M(n)$ be the number of moves required to move an n-disk Tower

Let $n = 1$. This takes just 1 move. → $M(1) = 1$

To move n disks we…

- Move n-1 disks to an auxiliary peg
- Move the bottom 1 disk to the end peg
- Move n-1 disks from auxiliary peg to end peg → $M(n) = 2\,M(n-1) + 1$

Our recurrence:

$$M(1) = 1$$

$$M(n) = 2\,M(n-1) + 1$$

Recursion – Towers of Hanoi

Our recurrence: $M(1) = 1,$ $M(n) = 2\,M(n\text{-}1) + 1$

Exploration:

$M(1) = 1$

$M(2) = 2(1) + 1 = 3$

$M(3) = 2(3) + 1 = 7$

$M(4) = 2(7) + 1 = 15$

$M(5) = 2(15) + 1 = 31$

…

Recursion – Towers of Hanoi

Our recurrence: $M(1) = 1$, $M(n) = 2\,M(n\text{-}1) + 1$

Exploration:

$M(1) = 1$

$M(2) = 2(1) + 1 = 3$

$M(3) = 2(3) + 1 = 7$

$M(4) = 2(7) + 1 = 15$

$M(5) = 2(15) + 1 = 31$

…

<u>Notice any patterns?</u>

Maybe… $M(n) = 2^n - 1$ (?)

Let's prove this using **induction**

Recursion – Towers of Hanoi

Base case: Let n=1. Then $M(1) = 2^1 - 1 = 1$

Induction step: Assume $M(l) = 2^l - 1$ for $1 \leq l \leq k$.

To show: $M(k+1) = 2^{k+1} - 1$

$$M(k+1) = 2\,M(k) + 1 \qquad \text{(recurrence)}$$

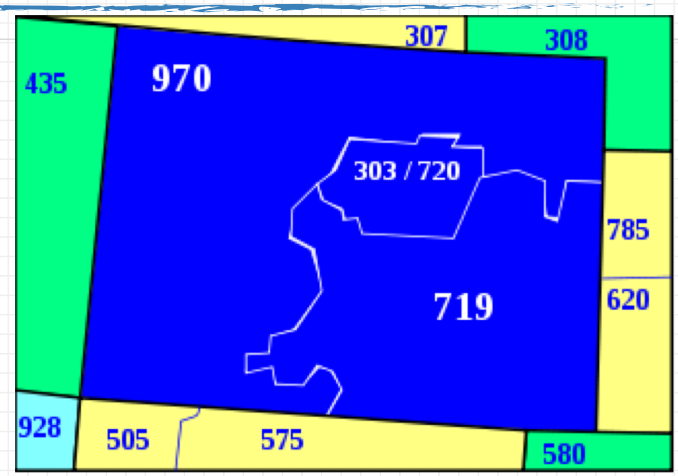
$$= 2\,(2^k - 1) + 1 \qquad \text{(inductive hypothesis)}$$

$$= 2^{k+1} - 2 + 1 \qquad \text{(maths)}$$

$$= 2^{k+1} - 1$$

Therefore, we've proved (by induction) that $M(n) = 2^n - 1$ ◇

Combinatorics



The art/science of counting things

Super important in all computational science

Answer questions like:

- Can a password of a particular form resist attack?
- Do we have enough “stuff” to meet some demand?
  - E.g., license plates, phone numbers (Boulder!), IP addresses
- What is the probability of a particular, ***discrete*** event occurring?

Counting basics

Many of the combinatorial concepts we'll talk about depend on two simple rules:

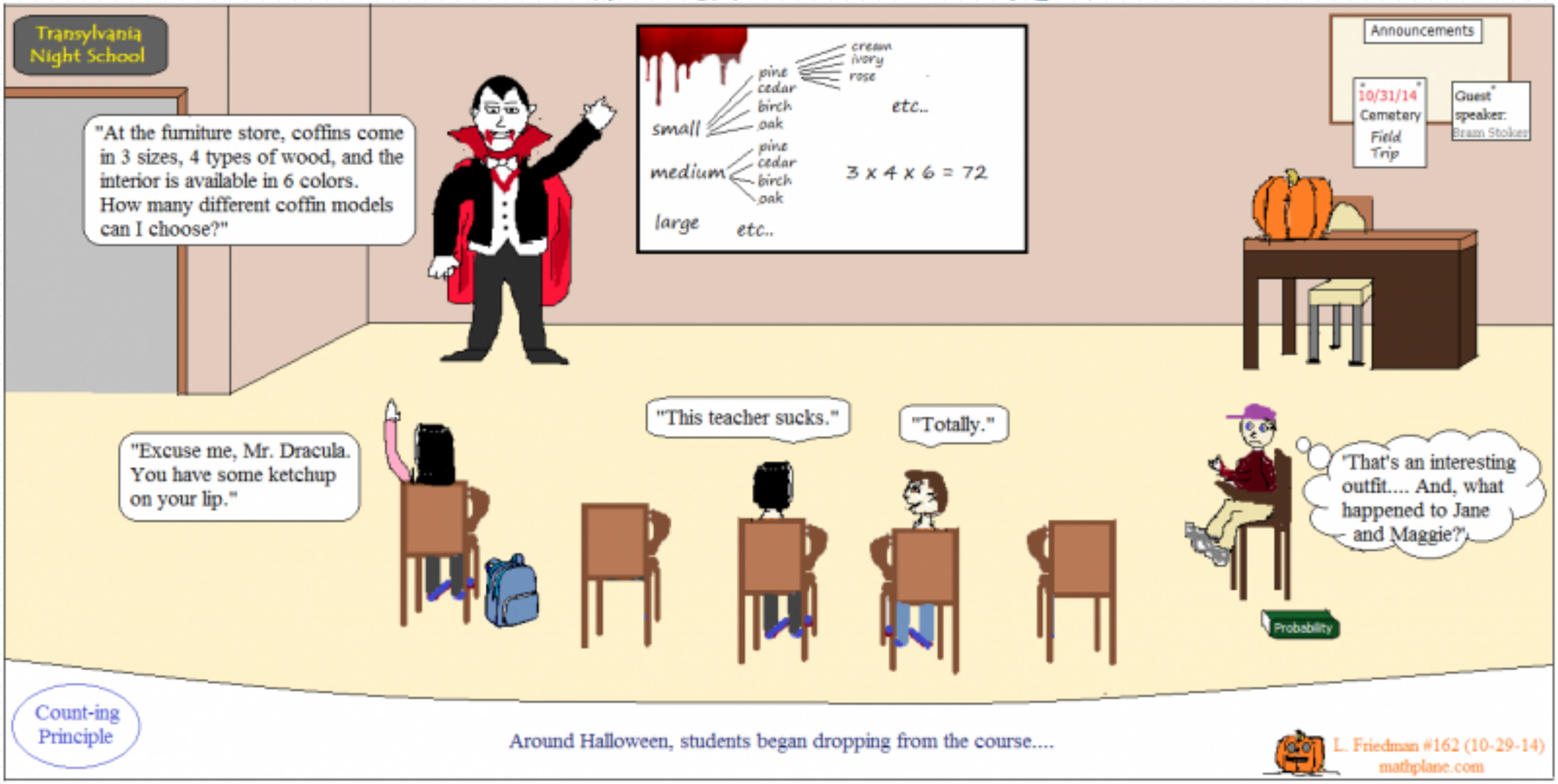
The Product Rule. S'pose that a procedure can be broken down into two tasks. If there are n_1 ways to do the first task, and for each of these ways to do the first task, there are n_2 ways to do the second task, then there are $n_1 \times n_2$ ways to do the whole procedure.

Shorthand: Number of ways to do both *Task 1* **and** *Task 2*

The Sum Rule. If a goal can be achieved in one of n_1 ways using *Task 1* or in n_2 ways using *Task 2* (where none of the n_1 or n_2 ways are the same), then there are $n_1 + n_2$ ways total to achieve the goal.

Shorthand: Number of ways to do *Task 1* **or** *Task 2*

The Product Rule



Counting basics

The Product Rule. S'pose that a procedure can be broken down into two tasks. If there are n_1 ways to do the first task, and for each of these ways to do the first task, there are n_2 ways to do the second task, then there are $n_1 \times n_2$ ways to do the whole procedure.

Example: S'pose Scooby can take two friends to the movies, and will take one male friend and one female friend. If Scooby has two male friends (Fred and Shaggy) and two female friends (Velma and Daphne), how many different groups might go to the movies?

Counting basics

The Product Rule.

Example: S'pose Scooby can take two friends to the movies, and will take one male friend and one female friend. If Scooby has two male friends (Fred and Shaggy) and two female friends (Velma and Daphne), how many different groups might go to the movies?

Pick the male friend:	then pick the female friend:	
Fred	Daphne	
	Velma	
Shaggy	Daphne	2 x 2 = 4 possible ways
	Velma	to choose the friends

Counting basics

Example: In an auditorium-style classroom, each seat is labeled by an uppercase letter (corresponding to the row) and a single digit (corresponding to the seat in the row). How many different possible seat labels are there?

Counting basics

Example: In an auditorium-style classroom, each seat is labeled by an uppercase letter (corresponding to the row) and a single digit (corresponding to the seat in the row). How many different possible seat labels are there?

There are 26 possible uppercase letters (rows)

There are 10 possible single digits (0-9)

$\Rightarrow$ So there are 26 x 10 = 260 possible seat labels

---

We can generalize the **product rule** beyond the choice of two *tasks*…

If a procedure requires m tasks and the k^{th} task can be done in n_k different ways, then there are

$n_1 \times n_2 \times n_3 \times \dots \times n_m$ different ways to do the procedure.

Counting basics

Example: Standard Colorado license plates consist of 3 numbers followed by 3 uppercase letters. How many distinct license plates can be made in this configuration?

Fun fact! →

REGISTERED VEHICLES IN COLORADO 2009

Automobiles	3,125,488
Buses	11,263
Trucks	1,026,790
Motorcycles	174,915
Other	685,689
Total	**5,024,145**

Source: Colorado Department of Revenue (2009)

Counting basics

Example: Standard Colorado license plates consist of 3 numbers followed by 3 uppercase letters. How many distinct license plates can be made in this configuration?

There are 10 possible numbers

There are 26 possible letters

$\Rightarrow$ This gives $10 \times 10 \times 10 \times 26 \times 26 \times 26 = 17{,}576{,}000$ possible CO license plates

Fun fact! →

REGISTERED VEHICLES IN COLORADO 2009

Automobiles	3,125,488
Buses	11,263
Trucks	1,026,790
Motorcycles	174,915
Other	685,689
Total	**5,024,145**

Source: Colorado Department of Revenue (2009)

Counting basics

Example: How many different bit-strings of length 7 are there?

Counting basics

Example: How many different bit-strings of length 7 are there?

Each bit in the string can be either 0 or 1 (2 options)

Since there are 7 "tasks" (choosing 0 or 1), with 2 options each, we have

$$2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 2^7 = 128 \text{ different bit-strings of length 7}$$

Counting basics

Example: How many *n*-length palindromic bit-strings are there?

For example: 101101 or 10101

Counting basics

Example: How many n-length palindromic bit-strings are there?

For example: 101101 or 10101

If n is even, then we're free to choose $n/2$ bits (just the first half of the bit-string)

E.g., the palindrome 1001 is defined entirely by just the first two bits, 10, because it is mirrored on the other side

$\Rightarrow$ Can be done in $2^{n/2}$ different ways

If n is odd, then we can choose the first $(n + 1)/2$ bits (get the left half, and the middle bit)

E.g. the palindrome 10101 is defined by the first 101 (length 3 = (5+1)/2))

$\Rightarrow$ Can be done in $2^{(n+1)/2}$ different ways

Counting basics

The Sum Rule. If a goal can be achieved in one of n_1 ways using *Task 1* or in n_2 ways using *Task 2* (where none of the n_1 or n_2 ways are the same), then there are $n_1 + n_2$ ways total to achieve the goal.

And extends naturally to more than 2 tasks:

Example: S'pose a student can pick a programming assignment from one of three subject areas. There are also lists of potential projects within each of the subject areas, containing 15, 13 and 17 projects, and none of the projects appear on multiple lists. How many possible projects are there to choose from?

Counting basics

The Sum Rule. If a goal can be achieved in one of n_1 ways using *Task 1* or in n_2 ways using *Task 2* (where none of the n_1 or n_2 ways are the same), then there are $n_1 + n_2$ ways total to achieve the goal.

And extends naturally to more than 2 tasks:

Example: S'pose a student can pick a programming assignment from one of three subject areas. There are also lists of potential projects within each of the subject areas, containing 15, 13 and 17 projects, and none of the projects appear on multiple lists. How many possible projects are there to choose from?

15 + 13 + 17 = **45 projects**

Sum/product rule: loop analogies

What is the value of `count`, where $n_1, n_2, \ldots, n_m$ are all positive integers?

Sum Rule: OR

```
count = 0
for i_1 in 1:n_1
   count += 1
for i_2 in 1:n_2
   count += 1
   …
for i_m in 1:n_m
   count += 1
```

Product rule: AND

```
count = 0
for i_1 in 1:n_1
   for i_2 in 1:n_2
          …
      for i_m in 1:n_m
         count += 1
```

Counting basics

What about using *both* the Product and Sum Rules?

Example: S'pose your system requires you to choose a password of between 6 and 8 characters. How many passwords are there if they can be made up of digits and uppercase letters, and each password must contain at least one digit?

Strategy: Count up all possible passwords, then subtract off the ones without any digits.

- 26 uppercase letters and 10 digits → 36 characters total
- 6, 7 or 8 characters in passwords

Counting basics

Strategy: Count up all possible passwords, then subtract off the ones without any digits.

- 26 uppercase letters and 10 digits → 36 characters total
- 6, 7 or 8 characters in passwords

Counting basics

Strategy: Count up all possible passwords, then subtract off the ones without any digits.

- 26 uppercase letters and 10 digits → 36 characters total
- 6, 7 or 8 characters in passwords

\# 6-char passwords = $36 \times 36 \times 36 \times 36 \times 36 \times 36 = 36^6$

And the number without any digits in them is $26 \times 26 \times 26 \times 26 \times 26 \times 26 = 26^6$

$\Rightarrow$ # legit 6-char passwords = $36^6 - 26^6$ = **1,867,866,560**

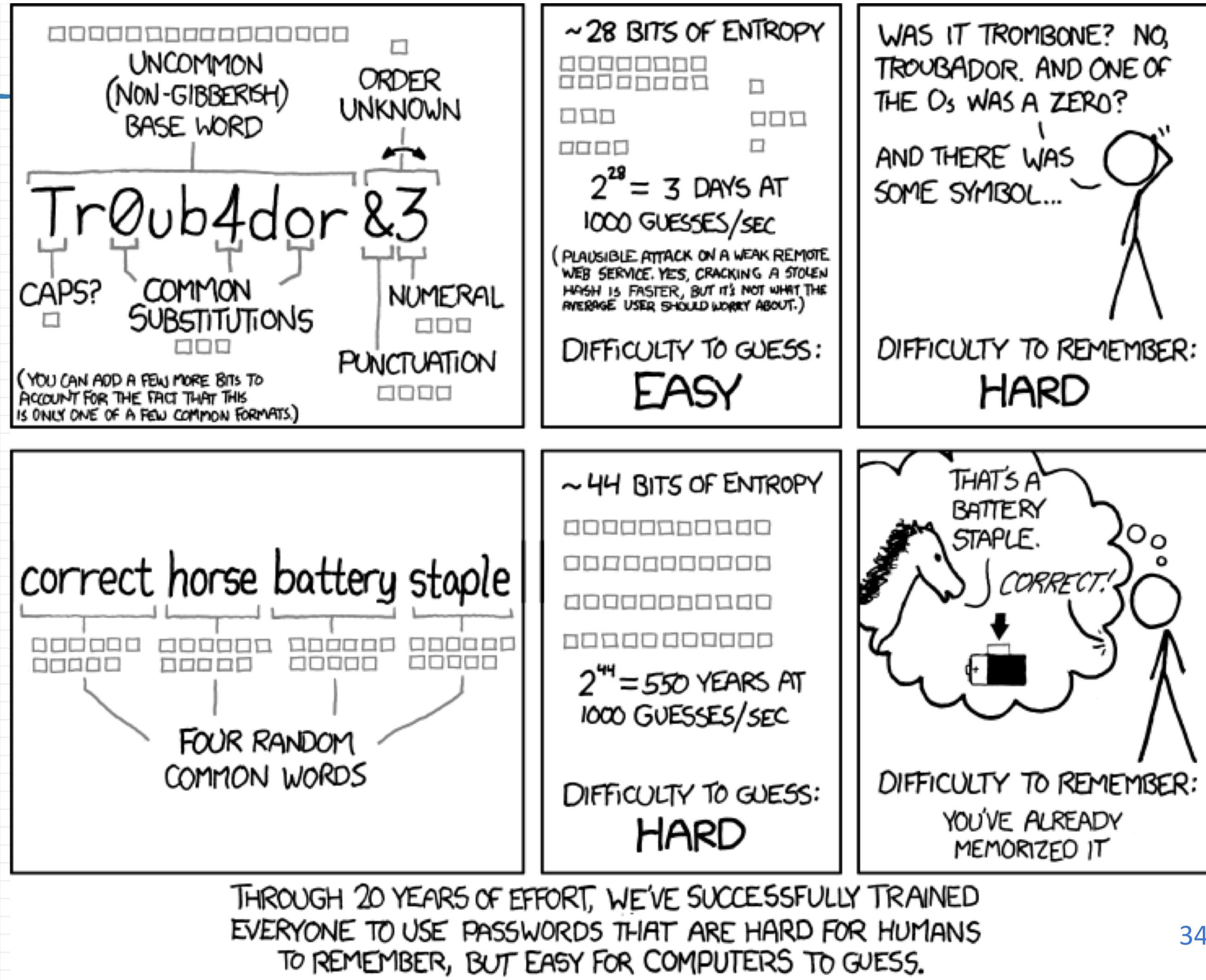
Similarly,

\# legit 7-char passwords = $36^7 - 26^7$ = **70,332,353,920**

\# legit 8-char passwords = $36^8 - 26^8$ = **2,612,282,842,880**

Total # legitimate passwords = about 2.7 trillion

And play around [here]() to see if this cartoon is accurate (in a crude way)



Counting basics

For the Sum Rule to apply, the possible sets of events need to be **distinct**. (non-overlapping)

Example: S'pose this summer, you want to take one class. You want it to be either one of 6 CS classes, or one of 4 APPM classes. Furthermore, one of the classes you're considering is cross-listed as both a CS course and an APPM course. How many possible classes could you take this summer?

Counting basics

For the Sum Rule to apply, the possible sets of events need to be **distinct**. (non-overlapping)

Example: S'pose this summer, you want to take one class. You want it to be either one of 6 CS classes, or one of 4 APPM classes. Furthermore, one of the classes you're considering is cross-listed as both a CS course and an APPM course. How many possible classes could you take this summer?

Answer = (6 CS classes) + (4 APPM classes) - (1 class that's both APPM and CS)
= **9 possible classes to take**

If this seems familiar, it should: Think of the *ways* to accomplish a task (or pick a class) as **sets** of events. Then we have

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

Counting basics

Example: How many different bit-strings of length 7 either begin with two 0s or end with three 1s?

Counting basics

Example: How many different bit-strings of length 7 either begin with two 0s or end with three 1s?

Begin with two 0s: 00xxxxx $\Rightarrow$ 5 tasks, 2 choices each $\Rightarrow$ 2^5 ways = 32

End with three 1s: xxxx111 $\Rightarrow$ 4 tasks, 2 choices each $\Rightarrow$ 2^4 ways = 16

But we need to subtract off the strings with **both** (the intersection):

Begin with two 0s **and** end with three 1s: 00xx111 $\Rightarrow$ 2^2 ways = 4

Total = 32 + 16 - 4 = **44**

FYOG: How many bitstrings of length 10 either begin with three 0s or end with two 0s?

Counting basics

Example: How many bit-strings of length 10 contain either five consecutive 0s or five consecutive 1s?

Counting basics

Example: How many bit-strings of length 10 contain either five consecutive 0s or five consecutive 1s?

6 positions that the 5 0s can start:

1. 00000xxxxx $\Rightarrow$ 2^5 choices
2. 100000xxxx $\Rightarrow$ 2^4
3. x100000xxx $\Rightarrow$ 2^4
4. xx100000xx $\Rightarrow$ 2^4
5. xxx100000x $\Rightarrow$ 2^4
6. xxxx100000 $\Rightarrow$ 2^4

Total from 0s $= 2^5 + 5 \cdot 2^4 =$ **112**

Counting basics

Example: How many bit-strings of length 10 contain either five consecutive 0s or five consecutive 1s?

6 positions that the 5 0s can start:

1. 00000xxxxx $\Rightarrow$ 2^5 choices
2. 100000xxxx $\Rightarrow$ 2^4
3. x100000xxx $\Rightarrow$ 2^4
4. xx100000xx $\Rightarrow$ 2^4
5. xxx100000x $\Rightarrow$ 2^4
6. xxxx100000 $\Rightarrow$ 2^4

Total from 0s $= 2^5 + 5 \cdot 2^4 =$ **112**

Same deal for the 1s: Total from 1s $=$ **112**

Counting basics

Example: How many bit-strings of length 10 contain either five consecutive 0s or five consecutive 1s?

6 positions that the 5 0s can start:

1. 00000xxxxx $\Rightarrow$ 2^5 choices
2. 100000xxxx $\Rightarrow$ 2^4
3. x100000xxx $\Rightarrow$ 2^4
4. xx100000xx $\Rightarrow$ 2^4
5. xxx100000x $\Rightarrow$ 2^4
6. xxxx100000 $\Rightarrow$ 2^4

Total from 0s = $2^5 + 5 \cdot 2^4$ = **112**

Same deal for the 1s: Total from 1s = **112**

But: 2 bitstrings with both:

0000011111 and 1111100000

Subtract those off to get our total:

Total = 112 + 112 - 2 = **222**

Counting basics

Example: How many positive integers not exceeding 100 are divisible by both 4 and 6?

Counting basics

Example: How many positive integers not exceeding 100 are divisible by both 4 and 6?

There are **25** integers divisible by 4:

$$1 \times 4,\ 2 \times 4,\ 3 \times 4, \ldots, 25 \times 4 \quad \leftarrow \text{because } 25 \times 4 = 100$$

Counting basics

Example: How many positive integers not exceeding 100 are divisible by both 4 and 6?

There are **25** integers divisible by 4:

$1 \times 4,\ 2 \times 4,\ 3 \times 4, \ldots, 25 \times 4$ ← because $25 \times 4 = 100$

… and there are **16** integers divisible by 6:

$1 \times 6, 2 \times 6, 3 \times 6, \ldots, 16 \times 6$ ← because $16 \times 6 = 96$

Counting basics

Example: How many positive integers not exceeding 100 are divisible by both 4 and 6?

There are **25** integers divisible by 4:

1×4, 2×4, 3×4, … , 25×4 ← because $25 \times 4 = 100$

… and there are **16** integers divisible by 6:

1×6, 2×6, 3×6, …, 16×6 ← because $16 \times 6 = 96$

… and there **8** integers divisible by both 4 and 6:

1×12, 2×12, … , 8×12 ← because $8 \times 12 = 96$, and 12 is the LCM of 4 and 6

Total = 25 + 16 - 8 = **33**

Counting basics

FYOG: How many positive integers between 100 and 999 (inclusive):

a) are divisible by 7
b) are odd
c) have the same three decimal digits
d) are not divisible by 4
e) are divisible by 3 or 4
f) are not divisible by either 3 or 4
g) are divisible by 3 but not 4
h) are divisible by 3 and 4

FYOG: How many strings of three decimal digits:

a) do not contain the same digit three times
b) begin with an odd digit
c) have exactly two digits that are 4s

Counting basics

Recap:

- Product rule -- when you have a sequence of tasks to complete, and each can be done in some number of different ways
- Sum rule -- when you have choices for *how* to complete a particular task

Next time:

- More counting, and
- How many ways are there to pick out some delicious donuts? (permutations and combinations)