CSCI 2200 FOUNDATIONS OF COMPUTER SCIENCE

David Goldschmidt
goldsd3@rpi.edu
Fall 2022

COUNTING SEQUENCES – SUM & PRODUCT RULES

Sum Rule. N objects of two types: N_1 of type-1 and N_2 of type-2. Then,

$$N = N_1 + N_2.$$

Let N be the number of choices for a sequence

$$x_1 x_2 x_3 \cdots x_{r-1} x_r.$$

Let N_1 be the number of choices for x_1;
Let N_2 be the number of choices for x_2 *after you choose* x_1;
Let N_3 be the number of choices for x_3 *after you choose* $x_1 x_2$;
Let N_4 be the number of choices for x_4 *after you choose* $x_1 x_2 x_3$;
⋮
Let N_r be the number of choices for x_r *after you choose* $x_1 x_2 x_3 \cdots x_{r-1}$.

$$N = N_1 \times N_2 \times N_3 \times N_4 \times \cdots \times N_r.$$

COMBINATIONS VS. PERMUTATIONS

Does order matter…?

We often want to select k objects from a set of n elements — sometimes, order matters

If order does not matter, we have a k-combination or k-subset

If order matters, we have a k-permutation or k-ordering

e.g., let $S = \{ a, b, c, d \}$ and let $k = 2$

How can we combine elements of S when order does not matter?

The 2-subsets are:

$\{ ab, ac, ad, bc, bd, cd \}$

How can we permutate elements of S when order does matter?

The 2-orderings are:

$\{ ab, ac, ad, ba, bc, bd, ca, cb, cd, da, db, dc \}$

COMBINATIONS VS. PERMUTATIONS

Does order matter…?

With $n = 10$ runners in a race…

How many possible top $k = 3$ finishes are there, i.e., how many ways can runners finish first (F), second (S), and third (T)?

$$n = |\{ FST \}| = 10 \times 9 \times 8 = 720$$

solution using the product rule

$$\frac{10!}{(10-3)!} = \frac{10!}{7!} = 10 \times 9 \times 8 = 720$$

solution using k-orderings formula

And how many possible ways can we pick $k = 3$ runners to finish in the top 3, i.e., we do not care what order they finish in?

We choose 3 from 10: $$\binom{10}{3} = \frac{n!}{k!(n-k)!} = \frac{10!}{3!(10-3)!} = \frac{10!}{3!7!} = \frac{10 \times 9 \times 8}{3 \times 2} = 120$$

solution using k-subsets formula

$(k_1, k_2, \ldots, k_r)$-ORDERINGS – FIXED GROUPS

Three different types of candy: (red, blue, green)

e.g., fix sequence length $n = 5$

— also fix the # of reds $k_1 = 3$, # of blues $k_2 = 1$, and # of greens $k_3 = 1$

how many (k_1, k_2, k_3)-orderings are there…?

$$\binom{5}{3} + \binom{5}{1} + \binom{5}{1} = 20 = \frac{5!}{3!1!1!}$$

$$\binom{n}{k_1, k_2, \ldots, k_r} = \frac{n!}{k_1! \times k_2! \times k_3! \cdots \times k_r!} = \frac{(k_1 + k_2 + \cdots + k_r)!}{k_1! \times k_2! \times k_3! \cdots \times k_r!}$$

$(k_1, k_2, \ldots, k_r)$-ORDERINGS – FIXED GROUPS

```
How many 10-bit binary sequences with four 1-bits are there?

n = 10    k  = 4    k  = 6    ==>     10!    =   10x9x8x7    =  ???
           1         2              -------     ----------
                                     4! 6!        4x3x2
```

$(k_1, k_2, \ldots, k_r)$-ORDERINGS

Given n letters, how many *anagrams* (i.e., distinct arrangements) are there…?

This is a k-ordering with $k = n$ — therefore, we have $n!$ anagrams

e.g., given $n = 3$ letters *ABC* — anagrams are *ABC*, *ACB*, *BAC*, *BCA*, *CAB*, *CBA*

What about when we have duplicate letters…?

e.g., given $n = 3$ letters OWW — anagrams are OWW, WOW, WWO

$$\binom{n}{k_1, k_2, \ldots, k_r} = \frac{n!}{k_1! \times k_2! \times k_3! \cdots \times k_r!} = \frac{(k_1 + k_2 + \cdots + k_r)!}{k_1! \times k_2! \times k_3! \cdots \times k_r!}$$

$(k_1, k_2, \ldots, k_r)$-ORDERINGS

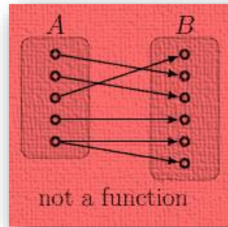

```
e.g., k-orderings for ABC (anagrams)

                          3!
 3-choose-(1,1,1)  =  --------  =  3!
                       1!1!1!

e.g., k-orderings for OWW (anagrams)

                        3!
 3-choose-(2,1)  =  ------  =  3
                     2!1!
```

SELECTING k FROM n DISTINGUISHABLE OBJECTS

Summary: selecting k objects from r types (colors):

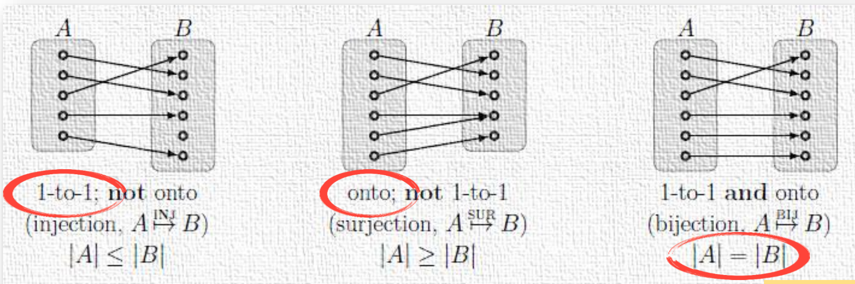
	no repetition	with repetition
k-sequence (order matters)	$\frac{r!}{(r-k)!}$	r^k
k-subset (order does not matter)	$\binom{r}{k}$	$\binom{k+r-1}{r-1}$
$(k_1, k_2, \cdots, k_r)$-sequence (k_1 of color-1,..., k_r of color-r)		$\binom{k_1+\cdots+k_r}{k_1, k_2, \ldots, k_r}$

BIJECTION, INJECTION, AND SURJECTION



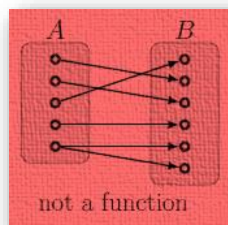
A function maps each element of set *A* to exactly one element of set *B*

We define *bijection*, *injection*, and *surjection* based on whether there is a *1-to-1 correspondence* or an *onto correspondence* — these are illustrated below:



See Pop Quiz 13.9…

BIJECTION, INJECTION, AND SURJECTION

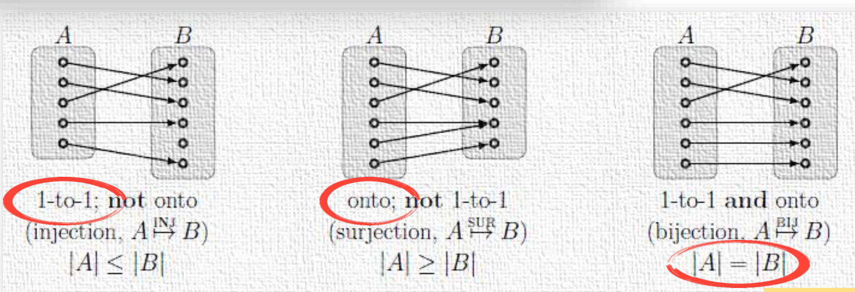


Clarifications:

- a 1-to-1 correspondence means every element in *A* maps to an element in *B*
- an onto correspondence means every element in *B* is mapped to by an element of *A*

A func… …ent of set *B*

We de… …ther there is a *1-to-1 correspondence* or an *onto correspondence* — these are illustrated below:



See Pop Quiz 13.9…

Can you also count how many ways we can place an opposing king and queen safely…?

BIJECTION — CHESS EXAMPLE

How many possible ways can we place a queen and king in different rows and columns…?

Consider the positions of the king (K) and queen (Q) on a standard chessboard…

Identify the king-queen position as sequence $c_K r_K c_Q r_Q$

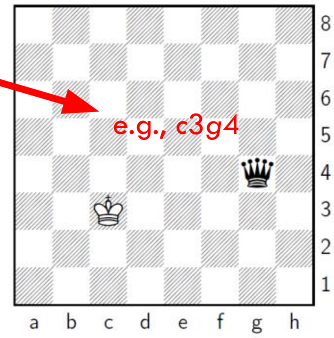
How can we count the number of *valid* sequences?

We need to count sequences with $r_K \neq r_Q$ and $c_K \neq c_Q$

Use the product rule — we have 8 choices for c_K and r_K…

…then 7 choices for c_Q and r_Q — multiply accordingly:

$$\#\text{ of sequences } = 8 \times 8 \times 7 \times 7 = 3136$$



Through bijection, the number of valid king-queen positions is 3136

Multiplicity Rule. If each object in A corresponds to k objects in B, then $|B| = k|A|$.

BIJECTION — CHESS EXAMPLE

How many possible ways can we place two rooks in different rows and columns…?

Consider the positions of two rooks (R_1 and R_2) on a standard chessboard…

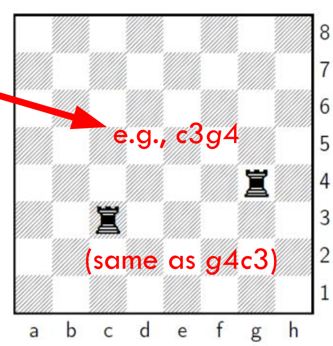
Identify the rook-rook position as sequence $c_{R_1} r_{R_1} c_{R_2} r_{R_2}$

How can we count the number of *valid* sequences?

Similar to our king-queen problem, but duplication occurs…

…because $c_{R_1} r_{R_1} c_{R_2} r_{R_2} = c_{R_2} r_{R_2} c_{R_1} r_{R_1}$ — we can divide by 2:

$$\#\text{ of sequences } = \frac{8 \times 8 \times 7 \times 7}{2} = \frac{3136}{2} = 1568$$



Through bijection and multiplicity, the number of valid rook-rook positions is 1568

INCLUSION-EXCLUSION PRINCIPLE

How many numbers in { 1, 2, …, 19, 20 } are divisible by 2 or by 5…?

Let set A = { numbers divisible by 2 } = { 2, 4, 6, …, 18, 20 } — therefore, $|A| = 10$

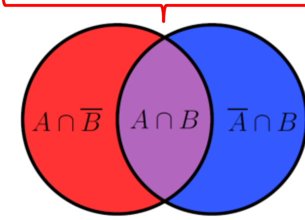
Let set B = { numbers divisible by 5 } = { 5, 10, 15, 20 } — therefore, $|B| = 4$

$|A|$ and $|B|$ are <u>not</u> disjoint sets, so how do we "un-count" duplicates 10 and 20…?

From the *inclusion-exclusion principle*, we have $|A \cup B| = |A| + |B| - |A \cap B|$

With $A \cap B = \{ 10, 20 \}$, we compute:

$$
\begin{aligned}
|A \cup B| &= |A| + |B| - |A \cap B| \\
&= 10 + 4 - 2 \\
&= 12 \checkmark
\end{aligned}
$$



How many numbers in { 1, 2, …, 1000 } are divisible by 2, by 3, or by 5…?

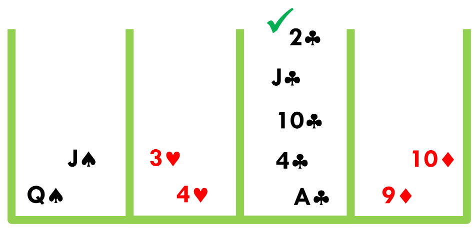
PIGEONHOLE PRINCIPLE

How many cards must we have to guarantee a 5-card flush (i.e., all the same suit)…?

Given 4 suits {♠, ♥, ♦, ♣}, we can create 4 corresponding bins for each card…

In this example, we got a 5-card flush after dealing 11 cards…

…but that was just luck (a.k.a., probability)



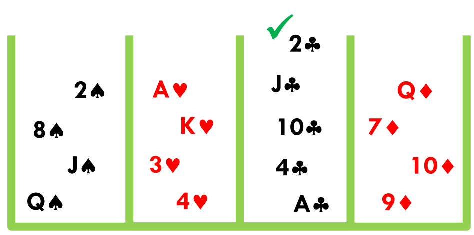
PIGEONHOLE PRINCIPLE

How many cards must we have to guarantee a 5-card flush (i.e., all the same suit)…?

Given 4 suits {♠, ♥, ♦, ♣}, we can create 4 corresponding bins for each card…

If we first maximize the number of cards in each bin, we guarantee that the next card drawn will give us a 5-card flush

The *pigeonhole principle* tells us that if we have more discrete objects than we have spaces (i.e., pigeonholes) for, at least one pigeonhole must exceed its capacity…



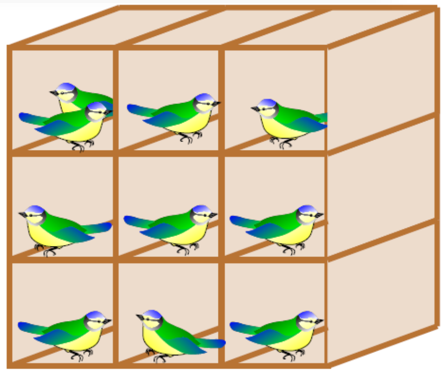
…so the answer is 17 cards

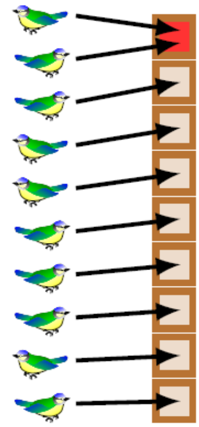
e.g., if you have eight people, at least two were born on the same day of the week

How many people do you need to ensure two are born on a Monday?

PIGEONHOLE PRINCIPLE

If there are *more* pigeons than pigeonholes, then at least one pigeonhole must have two or more pigeons.





We are just comparing set sizes here…

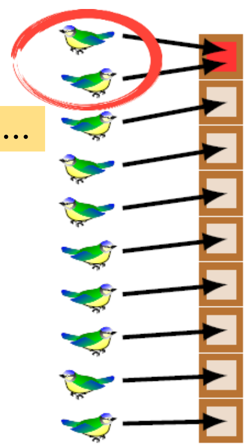
Let A = { pigeons } and B = { pigeonholes }

We would like to have $|A| \leq |B|$, i.e., if function $f: A \mapsto B$ is 1-to-1, then $|A| \leq |B|$

The contrapositive states that if $|A| > |B|$, then f cannot be 1-to-1 and we have overflow…

Given connected graph G, prove that two vertices must have the same degree…

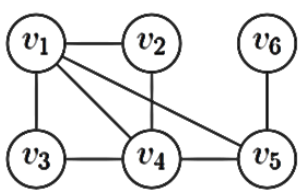
PIGEONHOLE PRINCIPLE



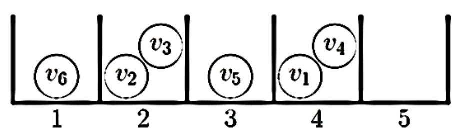
Given connected graph G, prove that two vertices must have the same degree…

Two vertices are *social twins* if they have the same degree

e.g., graph $G = (V, E)$ with $|V| = 6$ and $\partial = [4,4,3,2,2,1]$



$\Rightarrow$ pigeonholes represent degree values…



For $|V| = n$ vertices, degrees can be 1, 2, …, $(n - 1)$ — and no 0-degree vertices

Placing n pigeons (vertices) into $(n - 1)$ pigeonholes (degrees) requires at least one pigeonhole to contain two or more pigeons!

See Exercise 14.7…

WHAT NEXT…?

Look at the schedule and plan your last few weeks…!

Exam 2 grades have been posted along with a solutions PDF and video…

- …submit any grade inquiries by 11:59PM on Monday, December 5

Problem Set 8 has been posted…

- …and is due in your recitations on Wednesday, November 30

After Thanksgiving, we will covers Models of Computation, i.e., Chapters 23-29…

- …with Homework 5 due 11:59PM Thursday, December 8

* * * **Final Exam is scheduled 3:00-6:00PM on Wednesday, December 14** * * *

Practice! Tinker! Practice! Tinker! Practice! Tinker! Practice! Tinker! Practice!