Do whatever you want, but always explain what you are doing.

— Konstantin, 2020

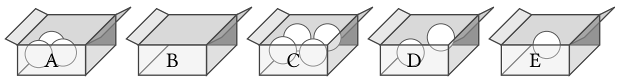
> The problems in this assignment are *tagged*:
>
> Core are essential problems that you *must solve* to pass the assignment.
>
> Challenge are *non-mandatory* problems that could be *skipped* for a passing grade.
>
> Bonus are *optional* problems for *memorable experience*. They won't be graded.

Problem 1: The Twelvefold Way — Core

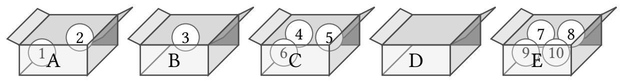
One of the classical combinatorial problems is counting the number of arrangements of n balls into k boxes. There are at least 12 variations of this problem: four cases (a–d) with three different constraints (1–3). For each problem (case+constraint), derive the corresponding generic formula. Additionally, pick several representative values for n and k and use your derived formulae to find the numbers of arrangements. Visualize some arrangements for your chosen values of n and k.

Cases with arrangement examples:

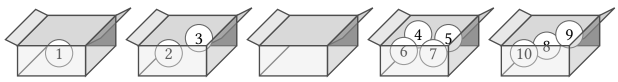
(a) $\boldsymbol{U} \to \boldsymbol{L}$: *Unlabeled* Balls, *Labeled* Boxes.



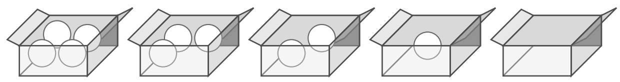
(c) $\boldsymbol{L} \to \boldsymbol{L}$: *Labeled* Balls, *Labeled* Boxes.



(b) $\boldsymbol{L} \to \boldsymbol{U}$: *Labeled* Balls, *Unlabeled* Boxes.



(d) $\boldsymbol{U} \to \boldsymbol{U}$: *Unlabeled* Balls, *Unlabeled* Boxes.



Constraints:

1. ≤ 1 ball per box — *injective* mapping.
2. ≥ 1 ball per box — *surjective* mapping.
3. Arbitrary number of balls per box.

Notes:

∗ ‣ *U*nlabeled means "indistinguishable", and *L*abeled means "distinguishable".

‣ Stirling number of the second kind $s^{\mathrm{II}}_k(n) = \left\{ {n \atop k} \right\} = S(n,k)$ is the number of ways to partition a set of n elements into k non-empty subsets. Use $s^{\mathrm{II}}_k(n)$ notation (or $\left\{ {n \atop k} \right\}$, or $S(n,k)$, to your preference) directly without expanding the closed formula.

‣ Partition function $p_k(n)$ is the number of ways to partition the integer n into k positive parts, *i.e.* the number of solutions to the following equation: $n = a_1 + \ldots + a_k$, where $a_1 \geq \ldots \geq a_k \geq 1$. Use $p_k(n)$ directly, since the closed-form expression is unknown.

Problem 2: Password Counting

Core

How many different passwords can be formed using the following rules?

- The password must be exactly 8 characters long.
- The password must consist only of Latin letters (a–z, A–Z) and Arabic digits (0–9).
- The password must contain at least 2 digits (0–9) and at least 1 uppercase letter (A–Z).
- Each character can be used no more than once in the password.

Find the minimum password length m (same alphabet and constraints) such that the number of valid passwords is at least 2^{128}. Also estimate what fraction of all 8-character strings over this alphabet are valid passwords under the given constraints.

Problem 3: Five-Digit Numbers

Core

Find the number of different 5-digit numbers using digits 1–9 under the given constraints. For each case, provide examples of numbers that comply and do not comply with the constraints, and derive a generic formula that can be applied to other values of n (total available digits) and k (number of digits in the number). Express the formula using standard combinatorial terms, such as k-combinations C_n^k and k-permutations $P(n, k)$.

(a) Digits *can* be repeated.

(b) Digits *cannot* be repeated.

(c) Digits *can* be repeated and must be written in *non-increasing*[1] order.

(d) Digits *cannot* be repeated and must be written in *strictly increasing* order.

(e) Digits *cannot* be repeated and the sum of the digits must be even.

(f) Digits *cannot* be repeated and *exactly two* of the five digits are even.

Problem 4: Combinatorial Identity

Core

Let n be a positive integer. Prove the following identity using a combinatorial argument:

$$\sum_{k=1}^{n} k \cdot C_n^k = n \cdot 2^{n-1}$$

Then verify the same identity by an algebraic derivation (for example, via $kC_n^k = nC_{n-1}^{k-1}$ or from the binomial theorem).

Problem 5: Vandermonde's Identity

Core

Let r, m, n be non-negative integers. Prove the following identity using a combinatorial argument:

$$\binom{m+n}{r} = \sum_{k=0}^{r} \binom{m}{k} \binom{n}{r-k}$$

[1] A sequence (x_n) is said to be *strictly monotonically increasing* if each term is *strictly greater* than the previous one, *i.e.* $x_i < x_{i+1}$. A sequence (x_n) is called *non-increasing* if each term is *less than or equal to* the previous one, *i.e.* $x_i \geq x_{i+1}$.

Problem 6: Generalized Pascal's Formula

Core

Prove the Generalized Pascal's Formula (for $n \geq 1$ and $k_1, ..., k_r \geq 0$ with $k_1 + ... + k_r = n$):

$$\binom{n}{k_1, ..., k_r} = \sum_{i=1}^{r} \binom{n-1}{k_1, ..., k_i - 1, ..., k_r}$$

Treat terms with $k_i = 0$ as zero (equivalently, omit those terms from the sum).

Problem 7: The Multinomial Walk

Core

A robot starts at $(0, 0, 0)$ in a 3D integer lattice. At each step it moves by one of the vectors $(1, 0, 0)$, $(0, 1, 0)$, or $(0, 0, 1)$.

(a) How many different paths of length 15 end at $(5, 7, 3)$?

(b) Generalize to endpoint (a, b, c) with $a + b + c = n$.

(c) Generalize further to d dimensions: endpoint $(a_1, ..., a_d)$ with $\sum_i a_i = n$.

Problem 8: Anagram Identities

Core

Consider the word ASSESSMENTS.

(a) Count the number of distinct anagrams.

(b) Count the number of distinct anagrams in which all five letters `S` appear consecutively.

(c) Generalize: if a multiset has size n with multiplicities $(m_1, ..., m_r)$, derive a formula for the number of permutations where all copies of one chosen symbol form a single consecutive block.

(d) If two distinct symbols are each required to form a consecutive block, explain whether adjacency between the two blocks changes the counting argument.

Problem 9: Non-Crossing Perfect Matchings

Challenge

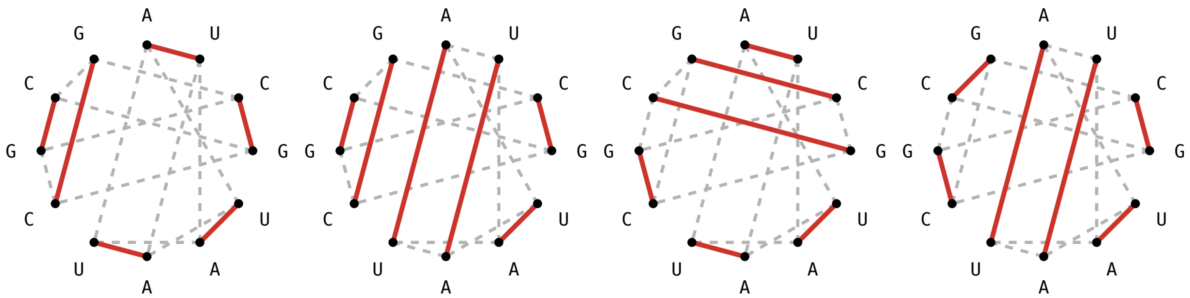
A *non-crossing perfect matching*[2] in a graph is a set of pairwise disjoint edges that cover all vertices and do not intersect with each other. For example, consider a graph on $2n$ vertices numbered from 1 to $2n$ and arranged in a circle. Additionally, assume that edges are straight lines. In this case, edges $\{i, j\}$ and $\{a, b\}$ intersect whenever $i < a < j < b$.

(a) Count the number of all possible non-crossing perfect matchings in a complete graph K_{2n}.

(b) Consider a graph on vertices labeled with letters from $\{`A`, `C`, `G`, `U`\}$. Each pair of vertices labeled with `A` and `U` is connected with a *basepair edge*. Similarly, `C`–`G` pairs are also connected.

The picture below illustrates some of possible non-crossing perfect matchings in the graph with 12 vertices `AUCGUAAUCGCG` arranged in a circle. Basepair edges are drawn dashed gray, matching is red.

[2] Credits to Rosalind for this task.



Let $s = \texttt{CGUAAUUACGGCAUUAGCAU}$ (20 vertices). Count the number of all possible non-crossing perfect matchings in the graph on vertices arranged in a circle and labeled with s.

Problem 10: Integer Solutions

Core

How many integer solutions are there for each given equation?

(a) $x_1 + x_2 + x_3 = 20$, where $x_i \geq 0$

(b) $x_1 + x_2 + x_3 = 20$, where $x_i \geq 1$

(c) $x_1 + x_2 + x_3 = 20$, where $x_i \geq 5$

(d) $x_1 + x_2 + x_3 \leq 20$, where $x_i \geq 0$

(e) $x_1 + x_2 + x_3 = 20$, where $1 \leq x_1 \leq x_2 \leq x_3$

(f) $x_1 + x_2 + x_3 = 20$, where $0 \leq x_1 \leq x_2 \leq x_3$

(g) $x_1 + x_2 + x_3 = 20$, where $0 \leq x_1 \leq x_2 \leq x_3 \leq 10$

(h) $x_1 + x_2 + x_3 = 5$, where $-5 \leq x_i \leq 5$

Problem 11: Dice Probability

Core

Consider three dice: one with 4 faces, one with 6 faces, and one with 8 faces. The faces are numbered 1 to 4, 1 to 6, and 1 to 8, respectively. Find the probability of rolling a total sum of 12.

Problem 12: Interesting Subsets

Challenge

Let $A = \{1, 2, 3, ..., 12\}$. Define an *interesting* subset of A as a subset in which no two elements have a difference of 3. Determine the number of interesting subsets of A.

Problem 13: People in a Line

Challenge

Let $f(n)$ be the number of ways to arrange n people of distinct heights in a line such that no three consecutive individuals form a strictly ascending or strictly descending height sequence. Compute $f(n)$ for $n = 1, 2, ..., 7$ and propose a conjecture (pattern, recurrence, or asymptotic trend).

Problem 14: GLaDOS Partitions

Challenge

GLaDOS, the mastermind AI, is testing a new batch of first-year students in one of her infamous test chambers. She assigns each test subject a unique number from 1 to n, and then splits the students into k indistinguishable groups. Furthermore, one student in each group is assigned as the group leader. GLaDOS wants to know how many different ways she can arrange the students into groups and select group leaders, so that the students can navigate through the test chambers without getting lost. She calls this arrangement a "GLaDOS Partition".

For example, consider $n = 7$ students and $k = 3$ groups. Here are three (out of many!) different partitions, with the group leaders underlined: $(\underline{1} \mid 2\underline{5}67 \mid 3\underline{4})$, $(\underline{1} \mid 25\underline{6}7 \mid \underline{3}4)$, and $(\underline{1} \mid 25\underline{6}7 \mid 3\underline{4})$.

Let the number of GLaDOS Partitions for n students into k groups, where each group has a designated leader, be denoted as $G(n, k)$. Your task is to find a generic formula and/or recurrence relation for $G(n, k)$ and justify it. Also compare $\sum_k G(n, k)$ with Bell numbers and explain the relation.

Problem 15: Stirling and Bell Synthesis

Core

Let $S(n, k)$ be the Stirling numbers of the second kind and let B_n be the n-th Bell number.

(a) Compute $S(5, k)$ for $k = 1, 2, 3, 4, 5$ and then compute B_5.

(b) Prove the identity $B_n = \sum_{k=1}^{n} S(n, k)$.

(c) Using $S(n, k) = S(n-1, k-1) + kS(n-1, k)$, compute B_6.

(d) Compare this problem with Problem 14 and express $G(n, k)$ in terms of $S(n, k)$.

* * *

Please make sure to answer *all* questions and provide *clear* explanations for your solutions.

Good luck!

Optional Bonus Problems

> The following problems are *"optional"*. They require programming and/or deeper theory. They will *not* count toward your grade but may earn bonus points and — more importantly — genuine understanding.

Problem A: Burnside's Lemma and Pólya Enumeration **Bonus**

Imagine you're coloring the faces of a cube with 3 colors. Naively, that's $3^6 = 729$ colorings — but if you rotate the cube, many of these look the same. How many are *truly* distinct?

This kind of question comes up all the time: distinct necklaces up to rotation, distinct molecules up to symmetry, distinct graph labelings up to automorphism. The key insight is that *symmetries form a group*, and you can count orbits by averaging over the group.

Burnside's Lemma. *Let a finite group G act on a set X. The number of distinct orbits (equivalence classes) is the average number of fixed points:*

$$|X \ / \ G| = \frac{1}{|G|} \sum_{g \in G} |\mathrm{Fix}(g)|$$

where $\mathrm{Fix}(g) = \{x \in X \mid g \cdot x = x\}$ *is the set of colorings left unchanged by* g.

In words: to count distinct objects, average over all symmetries how many configurations each symmetry preserves.

1. **Warm-up: binary bracelets.** You have n beads in a circle, each black or white. Two bracelets are the same if you can rotate one to get the other. Use Burnside's lemma to count distinct bracelets for $n = 4$ and $n = 5$. *(Try brute force first to check your answer.)*
2. **Necklaces with flips.** Now allow reflections too (you can flip the bracelet over).
   (a) How many distinct necklaces of $n = 6$ beads in $k = 3$ colors, up to rotation only?
   (b) Same question, but rotations and reflections are equivalent.
3. **Cube coloring.** How many ways to color the faces of a cube with k colors, up to rotation?

   The cube's rotation group has 24 elements. Your job is to figure out what types of rotations exist (identity, face rotations, edge rotations, vertex rotations), how many of each type, and how many colorings each type fixes.
4. **Pólya's theorem** (optional, for the adventurous). Instead of just counting total colorings, Pólya's theorem gives you a *generating function* that tells you how many colorings use each color how many times. For a cube colored with $\{R, G, B\}$, how many colorings use each color exactly twice?

Problem B: Gray Codes and Iterative Generation **Bonus**

Suppose you need to enumerate all 2^n bitstrings, or all C_n^k combinations, or all $n!$ permutations. You could just generate them in lexicographic order — but consecutive objects might differ in many positions. What if you want each step to change as *little* as possible?

This idea leads to *Gray codes* (flip one bit at a time), the *revolving door* algorithm (swap one element in/out), and the *Johnson–Trotter* algorithm (swap two adjacent elements). Knuth devotes an entire volume (TAOCP 4A) to these and related algorithms.

1. **Binary reflected Gray code.** The classic construction: $\Gamma_1 = (0, 1)$, and Γ_n is 0 prepended to Γ_{n-1}, followed by 1 prepended to the *reverse* of Γ_{n-1}. Implement this and verify for $n = 3, 4$ that consecutive strings differ in exactly one bit.
2. **Rank and unrank.** Can you go directly between a bitstring and its position in the Gray code, without generating the whole list? Implement $\operatorname{rank}(b) \to i$ and $\operatorname{unrank}(i) \to b$ in $O(n)$ time.
3. **Revolving door** (combinations). Generate all k-element subsets of $\{1, \ldots, n\}$ so that each subset differs from the previous one by exactly one element entering and one leaving — like a revolving door. Test for $n = 5, k = 3$.
4. **Johnson–Trotter** (permutations). Generate all $n!$ permutations so that each consecutive pair differs by swapping two *adjacent* elements. The trick: each element has a "direction" (left or right), and you always move the largest mobile element. Test for $n = 3, 4$.
5. **Benchmark.** Measure throughput (objects/second) for each algorithm: Gray code with $n \in \{10, 15, 20\}$, combinations with $n = 20, k = 10$, permutations with $n \in \{8, 9, 10\}$.

Problem C: Möbius Inversion and Derangements **Bonus**

You already know inclusion–exclusion: to count elements that have *none* of the bad properties, you alternate between overcounting and undercounting. It turns out this is a special case of something much more general — Möbius inversion — that works on *any* partially ordered set, not just subsets.

The idea: if you know a "cumulative" function $g(x) = \sum_{y \le x} f(y)$, you can recover f from g by inverting with the Möbius function of the poset.

> ***Möbius Inversion.*** *On a poset $(P, \le)$ with Möbius function μ, if $g(x) = \sum_{y \le x} f(y)$, then*
>
> $$f(x) = \sum_{y \le x} \mu(y, x) \cdot g(y)$$
>
> *Different posets give different μ, and thus different inversion formulas.*

1. **Warm-up: number theory.** On $\mathbb{Z}^+$ ordered by divisibility, the Möbius function is the classical $\mu(n)$ (the one from number theory). Use Möbius inversion to prove $\varphi(n) = \sum_{d \mid n} \mu(d) \cdot n / d$. Check by hand for $n = 12, 30$.
2. **Inclusion–exclusion is just a special case.** On the boolean lattice (subsets ordered by inclusion), show that $\mu(A, B) = (-1)^{|B| - |A|}$. Conclude: inclusion–exclusion is Möbius inversion on the subset lattice.

3. **Derangements: permutations with no fixed points.** A *derangement* is a permutation where nothing stays in place. The number of derangements of n elements is called $!n$ (subfactorial).
   (a) Derive $!n = n! \sum_{k=0}^{n} \frac{(-1)^k}{k!}$ via inclusion–exclusion.
   (b) Prove the recurrence $!n = (n-1) \cdot (!(n-1) + !(n-2))$ by a combinatorial argument. *(Hint: where does element 1 go? It has $n-1$ choices...)*
   (c) Show that, as a formal power series, the EGF of derangements is $e^{-x} \;/\; (1-x)$.
   (d) Prove that $!n \;/\; n! \to 1 \;/\; e$ as $n \to \infty$. Intuitively, if you shuffle a large number of cards — how likely is it that no card ends up in its original position?

4. **Partition lattice**. Partitions of $\{1, ..., n\}$ ordered by refinement form a lattice Π_n. Its Möbius function has a beautiful formula. Use it to derive the Stirling numbers $\left\{ {n \atop k} \right\}$ via Möbius inversion — connecting this problem back to Problem 1.

Problem D: Combinatorial Games and Sprague–Grundy **Bonus**

Two players, perfect information, no randomness, last move wins. Think Nim, or any game where you take turns and the one who can't move loses. It turns out *every* such game is secretly just Nim in disguise.

The key idea: each position has a *Grundy number* (a non-negative integer), and a position is losing iff its Grundy number is 0. To compute it, take the *mex* (minimum excludant) of all Grundy numbers reachable in one move. The Sprague–Grundy theorem says this completely solves the game.

Grundy number. $g(p) = \mathrm{mex}(\{g(s) \mid s \text{ is reachable from } p\})$, *where "mex" of a set of non-negative integers is the smallest non-negative integer* not *in the set.*

Key fact: $g(p) = 0$ *iff p is a losing position (the previous player wins). Everything else is winning.*

Composition: *if a game splits into independent sub-games, the combined Grundy number is the XOR of the parts.*

1. **Nim.** Several heaps of tokens. On your turn, remove any positive number of tokens from one heap. Prove Bouton's theorem: position $(h_1, ..., h_k)$ is losing iff $h_1 \oplus h_2 \oplus ... \oplus h_k = 0$.

2. **Subtraction games.** One heap of n tokens. You may remove $t \in T$ tokens per turn (say $T = \{1, 2, 3\}$). Compute Grundy numbers for $n = 0, ..., 20$. Notice the pattern? Now try $T = \{1, 4, 5\}$ — still periodic?

3. **Wythoff's game.** Two heaps. On your turn: remove any number from one heap, or the *same* number from both. This one is special — the P-positions involve the golden ratio φ.
   (a) Compute the P-positions for heap sizes up to $(20, 20)$. Do you see the pattern?
   (b) The P-positions turn out to be $(\lfloor k\varphi \rfloor, \lfloor k\varphi^2 \rfloor)$ for $k = 0, 1, 2, ...$, where $\varphi = \left(1 + \sqrt{5}\right) \;/\; 2$. Verify this matches your computation.
   (c) Why does the golden ratio show up? (This is a deep and beautiful question — even a partial answer is worth a lot.)

4. **Putting it together.** Suppose you're playing Nim with heaps $(7, 5, 3, 1)$. Is the first player winning? If yes, give a winning move. Now suppose the game is: one subtraction game with $T = \{1, 3, 4\}$ on a heap of 10, *plus* a Nim heap of 5. Who wins?