A bouquet of counting problems and word combinatorics

1 Election $\pm$ problem

Consider a closely fought election in West Bengal in April 2026 between two major political alliances, say Party A and Party B. As the counting of votes progresses constituency by constituency, television channels continuously display the running totals. Sometimes one party leads early, then the other catches up, and the lead may fluctuate dramatically throughout the day.

Now suppose Party A eventually wins by a very small margin: it receives $n+1$ constituencies while Party B receives n. A natural combinatorial question arises: In how many possible counting sequences does Party A remain always ahead of (or at least tied with) Party B during the entire counting process?

Example. For $n = 2$, there are five such sequences:

$$AAABB,\ AABAB,\ AABBA,\ ABAAB,\ ABABA.$$

Exercise 1.1:

Find out the number of such sequences for $n = 3, 4, 5$.

2 Well-formed formulas

Suppose p and q are variables and $*$ is a binary operation (you can think of addition or multiplication, for example). Here are some logical syntactic rules to form new expressions using the symbols $p, q, *$ and parentheses–the so-called *well-formed formulas*:

1. Every variable is a well-formed formula.
2. If s and t are well-formed formulas, then so is $(s * t)$.

Example.

1. Some examples of well-formed formulas: $p, q, (p * q), ((p * p) * q)$ etc.
2. Some non-examples of well-formed formulas: $(p * p) * q$, $(s * t$, $p * q$ etc. (In each case, think about *why* the expression is not a well-formed formula.)

Quick tests to determine if a given string of symbols is not a well-formed formula are the following.

Exercise 2.1:

1. The number of left parentheses '(' in a well-formed formula equals the number of right parentheses ')'.

2. (**Strong prefix property**) If s is a well-formed formula that is not a variable, and t is a proper left substring of s containing at least one symbol, then the number of left parentheses in t is strictly greater than the number of right parentheses in it.

Suppose you are tasked with arranging n pairs of parentheses while satisfying the **weak prefix property**: the number of left parentheses in any proper left substring is at least the number of right parentheses in it. For $n = 1$, you have only one choice, namely () but for $n = 2$, you have two choices, namely ()() or (()).

Example. The following arrangements of parentheses satisfy the weak prefix property but not the strong prefix property: (())(), ()(()()).

Exercise 2.2:

Find all expressions for $n = 3, 4, 5$ satisfying the weak prefix property. Can you relate this problem to Exercise 1.1?

Exercise 2.3:

Let C_n denote the number you counted in the above exercise. Is it possible to compute C_{n+1} if you know $C_0, C_1, \cdots, C_n$?

3 Building Mountains

Did you know that the tallest mountain in the world is Mauna Kea in Hawaii, and not Mount Everest, when measured from its base to peak?

Suppose you want to build a mountain range using n up-strokes (/) and n down-strokes (\) but you can never go below the sea level.

Example. For $n = 1$, you have only one choice: /\.

For $n = 3$, the landscape is more varied. The following are the valid mountain ranges with 3 up-strokes and 3 down-strokes: ///\\\, //\/\\, //\\/\, /\//\\, and /\/\/\.

Exercise 3.1:

List and count the distinct mountain ranges that you can build for $n = 5$.

You might want to think about parenthesis arrangements satisfying the weak prefix property when constructing mountain ranges.

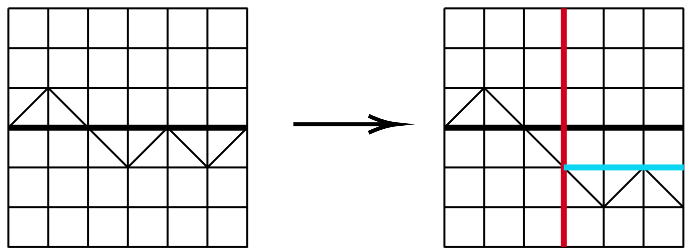
Counting mountain ranges using reflection (André's Method)

Let M_n denote the number of mountain ranges constructed using n-many of / and n-many \. The counting of M_n proceeds as follows: we will subtract the number of bad mountain ranges from the number of all possible mountain ranges, which can go below the sea level as well. Before we begin, let us look at a small example to illustrate the removal of bad mountain ranges:

Example. For $n = 2$, all (good and bad) mountain ranges are as follows: \\//, \/\/, /\\/, //\\, /\/\, \//\. Can you remove the bad mountain ranges from this list?

Step 1. What is the total number of mountain ranges using exactly n-many / and exactly n-many \?

Step 2. **The Reflection:** Look at the first instance where the mountain range drops below the sea level. We will reflect (flip / and \) the mountain range after that, as shown in the following figure.



Step 3. Note that the new mountain range has $(n-1)$-many / and $(n+1)$-many \. The reverse process of Step 2 can be applied to any mountain range having $(n-1)$-many / and $(n+1)$-many \ to get a bad mountain range.

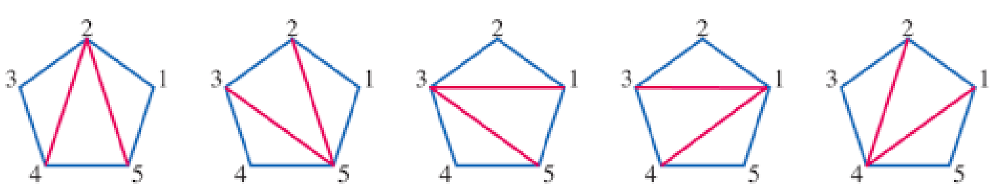
Step 4. What is the total number of mountain ranges having $(n-1)$-many / and $(n+1)$-many \?

Step 5. Subtract the number obtained in Step 1 from the number obtained in Step 4 to get the expression for M_n.

4 Triangulations of Polygons

Suppose you are lost in the Amazon forest but are holding a GPS-enabled device. Even if the forest contains cell towers only on its boundary, your location is determined by pinging three closest cell towers and calculating your distance from them–this method is called *triangulation*.

To understand the concept of triangulation mathematically, imagine a regular $(n+2)$-gon with vertices labeled 1 through $(n+2)$. Divide this polygon into exactly n triangles using $n-1$ non-intersecting diagonals. For a pentagon ($n = 3$), there are exactly five ways to achieve this.



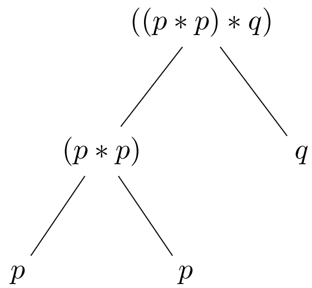
Exercise 4.1:

Count the number of triangulations of a (vertex-labeled) 6-gon and 7-gon. Is this problem related to Exercise 3.1? (If you find this second problem hard, you could revisit it after completing the next section.)

5 Full Binary Trees

Full binary trees are used by computers in the evaluation of well-formed formulas from Section 2–they are rooted trees where every node has either zero or two children. A node with zero children is called a *leaf node* and a node that is not a child of any other node is called the *root node*. An *internal node* is a node that is not a leaf node.

The following picture shows how the evaluation of $((p * p) * q)$ works.



Exercise 5.1:

Count the number of full binary trees with n internal nodes for $n = 3, 4$. Is this problem related to Exercise 2.2?

Exercise 5.2:

Is it possible to view the internal nodes of a full binary tree as triangles in a triangulation of an appropriate regular polygon? (The root could be the unique triangle containing the edge 12 in a given triangulation of the polygon.)

6 Bonus: Generating functions

Recall the numbers C_n from Exercise 2.3. Define the *generating function* of this sequence to be the expression $F(x) := \sum_{n \geq 0} C_n x^n$.

Example. A simple example of a generating function is that of the *geometric series*: $1 + x + x^2 + x^3 + \cdots = \frac{1}{1-x}$.

Simple manipulations of such expressions will actually turn out to be an easy way to evaluate sequences that satisfy certain recurrence relations. Here are some rules that will help us.

Exercise 6.1:

Let $A(x) = \sum_{n \geq 0} a_n x^n$. Fix a positive integer m.

1. Express $\sum_{n \geq m} a_{n-m} x^n$ in terms of $A(x)$.

2. Express $\sum_{n \geq 0} a_{n+m} x^n$ in terms of $A(x)$.

Exercise 6.2:

1. Let $A(x) = \sum_{n\geq 0} a_n x^n$ and $B(x) = \sum_{n\geq 0} b_n x^n$. Express $D(x) := \sum_{n\geq 0} (\sum_{k=0}^{n} a_k b_{n-k}) x^n$ in terms of $A(x)$ and $B(x)$.

2. Let $A_1(x), \ldots, A_m(x)$ be defined as $A_i(x) := \sum_{n\geq 0} a_{i,n} x^n$. Express $D(x) := \sum_{n\geq 0} (\sum_{k_1+\cdots+k_m=n} a_{1,k_1} a_{2,k_2} \cdots a_{m,k_m}) x^n$ in terms of $A_1(x), \ldots, A_m(x)$.

Exercise 6.3:

Use the recurrence relation obtained in Exercise 2.3 and the rules for manipulating generating functions that we saw above, show that $F(x) = 1 + xF(x)^2$. Use this to find an expression for $F(x)$ in terms of x.

7 Christoffel words

Given a positive integer n, a *Dyck path* (named after the German mathematician von Dyck, read *fon Daeek*) is a step-path from $(0, 0)$ to (n, n) that does not cross the line segment joining the end points. Dyck paths could be read as words over the alphabet $\{x, y\}$. For $n = 3$, there are 5 Dyck paths: $xxxyyy, xxyxyy, xyxxyy, xyxyxy, xxyyxy$.

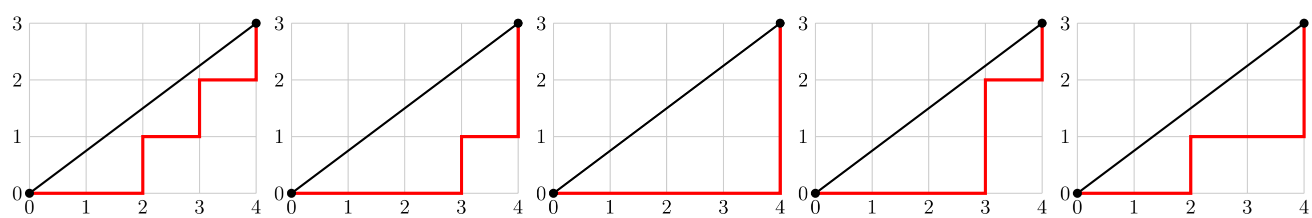
Exercise 7.1:

Is the counting of Dyck paths for given value of n related to any of the previous exercises?

Exercise 7.2:

Suppose there is a point (c, d) on the line segment joining $(0, 0)$ and (a, b), where a, b, c, d are all positive integers. What can you say about the relationship between a, b, c and d?

Now let us look at all step paths not crossing the line segment from $(0, 0)$ to (a, b), where a, b are positive integers. The following figure shows all such paths for $(a, b) = (4, 3)$.



(a) Path 1 (b) Path 2 (c) Path 3 (d) Path 4 (e) Path 5

Assume a, b are relatively prime. A *Christoffel path* is the step path that hugs the line segment, i.e., there is no point with integer coordinates in the region between the line segment and the step path. Only Path 1 in the above figure is a Christoffel path. The corresponding word $xxyxyxy$ is known as the *Christoffel word* of *slope* $\frac{3}{4} = \frac{\text{number of } y\text{s}}{\text{number of } x\text{s}}$.

Path 2 is not a Christoffel path since $(2, 1)$ lies in the area between the path and the line segment.

Exercise 7.3:

a) Find Christoffel words of slopes $\frac{3}{4}$, $\frac{2}{5}$ and $\frac{3}{2}$.
b) What can you say about first and last letter in a Christoffel word?
c) Can you guess the relationship between Christoffel words of slopes $\frac{a}{b}$ and $\frac{b}{a}$?
d) Show that there is a unique Christoffel word for a given (a, b).

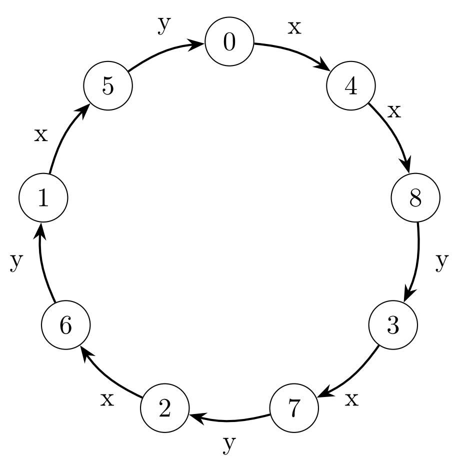
8 Christoffel words via Cayley graphs

Exercise 8.1:

Suppose a, b are relatively prime positive integers. Then show that a and $a + b$ are also relatively prime.

Consider the clock with $(a+b)$ hours labeled as $0, 1, \cdots, (a+b)-1$. Similar to our usual 12-hour clock, where 12 is replaced by 0, the clock-sum is defined as the remainder when the integer sum is divided by $(a + b)$. The Cayley graph $C(a + b; a)$ is drawn as follows: add an arrow from p to q if q is the clock-sum of p and a. We can label the arrow $p \to q$ by x or y according to whether the integer p is smaller than q or not.

Example. The following figure shows the labeled Cayley graph $C(9; 4)$. Starting at 0, read the labels along the direction of arrows to get the Christoffel word $xxyxyxyxy$ of slope $\frac{4}{5}$.



Exercise 8.2:

Find Christoffel words of slopes $\frac{3}{4}$ and $\frac{2}{5}$ using labeled Cayley graphs.

9 Perfectly clustering words

Given a word $w = a_1a_2 \cdots a_n$, for any $1 \leq i \leq n$, the word $w' = a_ia_{i+1} \cdots a_na_1 \cdots a_{i-1}$ is said to be a *cyclic permutation* of w.

Given a word w, its *Burrows-Wheeler transform* BWT(w) is defined as follows:

1. List all the cyclic permutations of w and arrange them as rows of a table (called the *Burrows-Wheeler table*), where any word in an upper row appears before any word in a lower row

in a dictionary. (Assume your dictionary contains all possibly nonsensical finite words with English alphabet!)

2. Read the last column from top to bottom to get the Burrows-Wheeler transform BWT(w).

Example. The Burrows-Wheeler table for the word banana is shown below.

a b a n a n
a n a b a n
a n a n a b
b a n a n a
n a b a n a
n a n a b a

Therefore, $\mathrm{BWT}(banana) = nnbaaa$.

Exercise 9.1:

1. Find the Burrows-Wheeler transform of the following words: carrot, tomato.
2. If a word w' is a cyclic permutation of the word w, then show that $\mathrm{BWT}(w) = \mathrm{BWT}(w')$.

A word w is said to be *perfectly clustering* if $\mathrm{BWT}(w) = b_1 b_2 \cdots b_n$ is *non-increasing*, i.e., b_i does not appear later in the dictionary than b_j for any $1 \leq i < j \leq n$.

Example. The word banana is a perfectly clustering word.

Exercise 9.2:

1. Find all perfectly clustering words of length at most 5 using letters x and y.
2. Can you find any pattern in the solutions to the above problem?