MATH 239 Fall 2022:
Lecture Notes

Lecture notes taken, unless otherwise specified, by myself during section 002 of the Fall 2022 offering of MATH 239, taught by Luke Postle.

Part/chapter titles vaguely follow the official course notes.

Part I

Enumeration

Chapter 1

Introduction

Lecture 1 (09/07; Section 004)

By convention, define the natural numbers $\mathbb{N} = \{0, 1, 2, \dots\}$. If we want the non-negative integers, we can write $\mathbb{N}_{\geq 1}$.

Recall from set theory, for sets A and B, the union $A \cup B = \{c : c \in A \vee c \in B\}$ and intersection $A \cap B = \{c : c \in A \wedge c \in B\}$. We call a union $A \cup B$ disjoint if $A \cap B = \varnothing$. When considering the sizes of finite sets, the disjoint union corresponds to addition: $|A \cup B| = |A| + |B|$.

Recall also the Cartesian product $A \times B = \{(a, b) : a \in A, b \in B\}$. Again for finite sets, $|A \times B| = |A| \cdot |B|$.

Definition (*bijection*)

A function $f : A \to B$ such that f is injective (i.e., $f(a) = f(a') \implies a = a'$) and surjective (i.e., $\forall b, \exists a, f(a) = b$). When f exists, we write $A \rightleftharpoons B$.

Theorem (*Bijective Proofs*)

If $A \rightleftharpoons B$, then $|A| = |B|$.

Proof. Let $f : A \to B$ be a bijection. For each element of B, it is the image of at least one element of A under f (by surjectivity) and at most one element (by injectivity). Therefore, A contains the same number of elements as B. $\square$

Lecture 2 (09/09)

By convention, let $[n] := \{1, \dots, n\}$.

Theorem

Let A be the set of subsets of $[n]$. Let B be the set of binary strings of length n. There exists a bijection from A to B.

Proof. Let $f : A \to B$ be defined as $f(S) = a_1 \dots a_n$ where $a_i = 1$ if $i \in S$ and 0 otherwise. Then, $f^{-1}(a_1 \dots a_n) = \{i \in [n] : a_i = 1\}$. Since $f^{-1} \circ f = \mathrm{id}$ and $f \circ f^{-1} = \mathrm{id}$, we have a bijection. $\square$

Theorem

Let A be the set of subsets of $[n]$ of size exactly k. Let B be the set of binary strings of length n with exactly k ones. There exists a bijection from A to B.

Proof. Restrict the domain of f from above. □

Definition (*permutation*)

A list of $[n]$ for some positive integer n. That is, a bijection from $[n]$ to $[n]$. Notate the set of permutations of $[n]$ by P_n.

Theorem (*1.3*)

The number of subsets of $[n]$ is 2^n.

Proof. $\mathcal{P}([n]) \rightleftharpoons \{0,1\}^n$ by the above theorem. But we know $|\{0,1\}^n| = |\{0,1\}|^n = 2^n$. Then, $|\mathcal{P}([n])| = 2^n$. □

Theorem (*1.5*)

The number of subsets of $[n]$ of size k is $\binom{n}{k}$.

Proof. Let $S_{n,k}$ be the subsets of $[n]$ of size k.

For each $A \in S_{n,k}$ define $P_A := \{\sigma_1 \dots \sigma_n \in P_n : \{\sigma_1, \dots, \sigma_n\} = A\}$. Then, P_n is the disjoint union of the sets P_A. Also, $P_A \rightleftharpoons P_k \times P_{n-k}$ because the first k entries are a list of A and the last $n-k$ entries are a list of $[n] \setminus A$.

Therefore, $|P_A| = k! \cdot (n-k)!$ and $\left|S_{n,k}\right| = \frac{|P_n|}{k!\cdot(n-k)!} = \binom{n}{k}$ as the sets are of equal size. □

In general, to prove that A has some size, we can either

(1) Prove A is a disjoint union of smaller sets of known size
(2) Prove A is a Cartesian product of smaller sets of known size
(3) Give a bijection $A \rightleftharpoons B$ to a set of known size
(4) Show a family of sets A_i satisfies a recurrence relation and use induction

Definition (*composition*)

A finite sequence $n = (m_1, \dots, m_k)$ of positive integers called parts. The size of the composition $|n| = \sum m_i$.

Theorem

For all $n \geq 1$, there are 2^{n-1} compositions of size n

Proof. Proceed by induction on n.

When $n = 1$, there is exactly one composition (1). Assume $n \geq 2$.

Let A_n be compositions of size n. Also let $B_{1,n} := \{(a_1, \dots) \in A_n : a_1 = 1\}$ and $B_{2,n} := \{(a_1, \dots) \in A_n : a_1 \geq 2\}$. Notice that since a_1 is a positive integer so either $a_1 = 1$ or $a_1 \geq 2$, so A_n is the disjoint union of $B_{1,n}$ and $B_{2,n}$.

Let $f : B_{1,n} \to A_{n-1}$ be given by $f((b_1, \dots, b_k)) = (b_2, \dots, b_k)$ and we can find the inverse $f^{-1}((a_1, \dots, a_k)) = (1, a_1, \dots, a_k)$.

Let $g : B_{2,n} \to A_{n-1}$ be given by $g((b_1, \dots, b_k)) = (b_1 - 1, b_2 \dots, b_k)$ whose inverse we can likewise find $g^{-1}((a_1, \dots, a_k)) = (a_1 + 1, a_2, \dots, a_k)$ assuming A_{n-1} is nonempty, i.e., $n \geq 2$.

As these are bijections, $|B_{1,n}| = |n-1| = |B_{2,n}|$. By induction $|A_{n-1}| = 2^{n-2}$ and hence $|A_n| = |B_{1,n}| + |B_{2,n}| = 2|A_{n-1}| = 2^{n-1}$ as desired. □

Lecture 3 (09/12)

Theorem (*Binomial Theorem*)

For all $n \geq 1$, $(1+x)^n = \sum_{k=0}^{n} \binom{n}{k} x^k$

Proof. Must choose either 1 or x from each monomial, so we have $(1+x)^n = \sum_{S \subseteq [n]} x^{|S|} = \sum_{k=0}^{n} \binom{n}{k} x^k$ by grouping choices by the number of x chosen and applying Theorem 1.5. □

Corollary. If $x = 1$, then $2^n = \sum_{k=0}^{n} \binom{n}{k}$, i.e., Theorem 1.3.

Proof. From Theorem 1.2, $|\mathcal{P}([n])| = 2^n$. But $|\mathcal{P}([n])| = \left|\bigcup S_{n,k}\right| = \sum |S_{n,k}| = \sum \binom{n}{k}$ by Theorem 1.5. □

Corollary. If $x = -1$, then $0 = \sum_{k=0}^{n} \binom{n}{k} (-1)^k$.

Proof. See Tutorial 1-2. □

Definition (*multiset of t types*)

Sequence $a_1, \ldots, a_t$ where each a_i is a non-negative integer. The a_i are the *parts* and the sum $\sum a_i$ is the *size*.

Like compositions but permitting zero and restricting $i = 1, \ldots, t$. For example, consider the multiset $(3, 2, 0, 1)$ as being like a "set" $\{a_1, a_1, a_1, a_2, a_2, a_4\}$.

Theorem (*1.9*)

The number of multisets of t types and size n is $\binom{n+t-1}{t-1}$.

Proof. Consider the creation of a multiset like dividing up the line of elements (e.g. $(3, 2, 0, 1)$ as $* * * \mid ** \mid \mid *$).

Encode this as a binary string where 0 means take an element and 1 means switch to the next set (in this case, 000100110). This is a string of length $n + t - 1$ with exactly $t - 1$ ones. That is, a subset of $[n + t - 1]$ with size $t - 1$, of which there are $\binom{n+t-1}{t-1}$.

To be formal, write out and prove that the set of multisets A bijects with $B = S_{n+t-1,t-1}$.

Under $f : A \to B : (a_1, \ldots, a_t) \mapsto \{a_1 + 1, \ldots, a_{t-1} + 1\}$ with inverse $f^{-1} : B \to A : \{b_1, \ldots, b_{n-1}\} \mapsto (b_1 - 1, b_2 - b_1 - 1, b_3 - b_2 - 1, \ldots, n - b_{t-1} - 1)$, $A \rightleftharpoons B$ and we have $|A| = \binom{n+t-1}{t-1}$, as desired. □

Theorem (*Negative Binomial Series*)

$\frac{1}{(1-x)^t} = \sum_{t=0}^{\infty} \binom{n+t-1}{t-1} x^n$

Proof. Notice that $\frac{1}{(1-x)^t} = \underbrace{(1 + x + x^2 + \cdots) \cdots (1 + x + x^2 + \cdots)}_{t \text{ times}}$. The coefficient of x^k is the number of multisets $(\alpha_1, \ldots, \alpha_t)$ of k size so apply Theorem 1.9. □

Lecture 4 (09/14)

Example 4.1 (Pascal's Identity)**.** $\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$

Proof (informal). We are counting $S \subseteq [n]$. There are two kinds of subsets: $n \in S$ and $n \notin S$. If $n \in S$, then $S \setminus \{n\} \subseteq [n-1]$ with size $n-1$, i.e., there are $\binom{n-1}{k-1}$ of these. If $n \notin S$, then $S \subseteq [n-1]$, i.e., there are $\binom{n-1}{k}$ of these. ☐

Proof. Let $S_{n,k} := \{S \subseteq [n] : |S| = k\}$ be the set of subsets of $[n]$ with size exactly k. By Theorem 1.5, $|S_{n,k}| = \binom{n}{k}$.

Let $A_1 = \{S \in S_{n,k} : n \in S\}$ and $A_2 = \{S \in S_{n,k} : n \notin S\}$. Then, $S_{n,k}$ is the disjoint union of A_1 and A_2 since either $n \in S$ or $n \notin S$. Thus, $|S_{n,k}| = |A_1| + |A_2|$.

Claim $A_1 \rightleftharpoons S_{n-1,k-1}$ under $f : A_1 \to S_{n-1,k-1} : S \mapsto S \setminus \{n\}$ with inverse $f^{-1} : S_{n-1,k-1} \to A_1 : T \mapsto T \cup \{n\}$. Then, $|A_1| = |S_{n-1,k-1}| = \binom{n-1}{k-1}$.

Claim $A_2 \rightleftharpoons S_{n-1,k}$ under $g = \mathrm{id}_{A_2}$ with $g^{-1} = \mathrm{id}_{S_{n-1,k}}$. Then, $|A_2| = |S_{n-1,k}| = \binom{n-1}{k}$.

Finally, $\binom{n}{k} = |S_{n-1,k}| = \binom{n-1}{k} + \binom{n-1}{k-1}$. ☐

Example 4.2. $\binom{n}{k} = \sum_{i=k}^{n} \binom{i-1}{k-1}$ for all $n \geq k \geq 1$

Proof (informal). Subsets of $[n]$ of size exactly k come in $n-k+1$ types, namely, classify by their maxima. Then, if i is largest, we have to chose $k-1$ remaining elements from $[i-1]$. This goes down to k being largest by the pigeonhole principle where we have $k-1$ elements from $[k-1]$. ☐

Proof. By Theorem 1.5, $|S_{n,k}| = \binom{n}{k}$.

Then, since the maximum of a set is unique, we can partition $S_{n,k}$ into the disjoint union $A_k \cup \cdots \cup A_n$ where for each i, $A_i = \{S \in S_{n,k} : \max S = i\}$. Thus, $|S_{n,k}| = \sum_{i=k}^{n} |A_i|$

Claim $A_i \rightleftharpoons S_{i-1,k-1}$ under the bijection $f : A_i \to S_{i-1,k-1} : S \mapsto S \setminus \{i\}$ with inverse $f^{-1} : S_{i-1,k-1} \to A_i : T \mapsto T \cup \{i\}$ from above. Then, $|A_i| = \binom{i-1}{k-1}$.

Finally, $\binom{n}{k} = |S_{n,k}| = \sum_{i=k}^{n} \binom{i-1}{k-1}$, as desired. ☐

Chapter 2

Generating Series

Lecture 5 (09/16)

Definition (*weight function*)

$w : A \to \mathbb{N}$ such that $\left|\text{preim}_w(n)\right|$ is finite for all $n \in \mathbb{N}$. Call $w(a)$ the weight of a.

In general, answer questions of the form "how many elements of A have weight n?"

Definition (*generating series*)

$\Phi_A^w(x) = \sum_{a \in A} x^{w(a)}$ for a set A and weight function w.

This is not a polynomial but a formal power series with infinite terms. We do not ever evaluate $\Phi_A^w(x)$ so convergence is irrelevant.

Basically, take the sequence $(\left|\text{preim}_w(0)\right|, \left|\text{preim}_w(1)\right|, \dots)$ and make it into coefficients.

Notate $[x^n]f(x)$ for the coefficient of x^n in $f(x)$.

Example 5.1. For $A = \mathcal{P}([n])$, $w = |\cdot|$, we have $\Phi_A^w(x) = \sum_{k=0}^n \binom{n}{k} x^k = (1+x)^n$

Proposition

The number of elements of weight i is the ith coefficient in $\Phi_A^w(x)$. Equivalently, $\Phi_A^w(x) = \sum_{n \geq 0} a_n x^n$ where $a_n = \left|\text{preim}_w(n)\right|$.

Example 5.2. For set of multisets with t types A and w size of the multiset, $\Phi_A^w(x) = \sum_{n \geq 0} \binom{n+t-1}{t-1} x^n = \frac{1}{(1-x)^t}$.

Example 5.3. For set of binary strings A and w length, $\Phi_A^w(x) = \sum_{n \geq 0} 2^n x^n = \sum_{n \geq 0} (2x)^n = \frac{1}{1-2x}$.

Example 5.4. For set of compositions A and w size, $\Phi_A^w(x) = 1 + x + 2x^2 + 4x^3 + \cdots = 1 + x(1 + 2x + 4x^2 + \cdots)$. This is $1 + x(\frac{1}{1-2x}) = \frac{1-x}{1-2x} = \frac{1}{1-\frac{x}{1-x}}$.

Definition (*formal power series*)

$A(x) = \sum_{n \geq 0} a_n x^n$ where a_n is finite for all $n \in \mathbb{N}$

We define addition, subtraction, and equality of formal power series akin to polynomials: $[x^n](A(x) + B(x)) = a_n + b_n$, $[x^n](A(x) - B(x)) = a_n - b_n$, $A(x) = B(x) \iff \forall n (a_n = b_n)$.

Lecture 6 (09/19)

Define multiplication of formal power series: $A(x)B(x) = \sum_{n\geq 0}(\sum_{k=0}^{n} a_k b_{n+k})x^n$.

Define division using inverses: $\frac{1}{A(x)}$ is the power series $C(x)$ such that $A(x)C(x) = 1$ where we consider the unit power series $1 + 0x + 0x^2 + \cdots$.

If $A(x)C(x) = 1$, then each coefficient $\sum_{k=0}^{n} a_k c_{n-k} = 0$ for $n \geq 1$ and $a_0 c_0 = 1$.

Solving, $c_0 = \frac{1}{a_0}$, $c_1 = -\frac{a_1 c_0}{a_0} = -\frac{a_1}{a_0^2}$, $c_2 = -\frac{1}{a_0}(a_1 c_1 - a_2 c_0)$, etc.

In general, $c_0 = \frac{1}{a_0}$ and $c_n = -\frac{1}{a_0}\sum_{k=0}^{n-1} a_k c_{n-k}$ for $n \geq 1$.

Theorem

$A \in \mathbb{R}[[x]]$ has an inverse $C = A^{-1}$ if and only if $a_0 \neq 0$. If $C(x)$ exists, $c_0 = a_0^{-1}$ and $c_n = -a_0^{-1}\sum_{k=1}^{n} a_k c_{n-k}$.

Given these four operations, we have the ring $\mathbb{R}[[x]]$ of power series. This is a superring of the polynomials $\mathbb{R}[x]$.

Theorem

$A \circ B \in \mathbb{R}[[x]]$ exists if and only if $A \in \mathbb{R}[x]$ or $b_0 = 0$

Lecture 7 (09/21)

Lemma (*Sum Lemma*)

Let C be the disjoint union $A \cup B$. Let w be a weight function of C. Then, $\Phi_C^w(x) = \Phi_A^w(x) + \Phi_B^w(x)$.
Note: Since $w : C \to \mathbb{N}$, we implicitly let $\Phi_A^w = \Phi_A^{w|_A}$ and $\Phi_B^w = \Phi_B^{w|_B}$

Proof. We will show equality by the coefficient definition, i.e., $[x^n]\Phi_C^w(x) = [x^n](\Phi_A^w(x) + \Phi_B^w(x)) = [x^n]\Phi_A^w(x) + [x^n]\Phi_B^w(x)$. The LHS is just the number of elements in $C = A \cup B$ of weight n and the RHS is the sum of the elements in A and B of weight n. These must be equal because $A \cap B = \varnothing$. □

Proof. Expand:

$$\Phi_C^w(x) = \sum_{c\in C} x^{w(c)} = \sum_{a\in A} x^{w(a)} + \sum_{b\in B} x^{w(b)} = \Phi_A^w(x) + \Phi_B^w(x)$$

where we can divide the sum because every $c \in C$ is in exactly one of A or B. □

Lemma (*Infinite Sum Lemma*)

Let C be the disjoint union $\bigcup A_i$ of countably infinite sets. Let w be a weight function of C. Then, $\Phi_C^w(x) = \sum \Phi_{A_i}^w(x)$.

Proof. Since w is a weight function, the preimage is finite. That is, $[x^n]\Phi_C^w(x)$ is finite and we can decompose the finite set $\{c \in C : w(c) = n\}$ into finitely many $\{c \in A_i : w(c) = n\}$, which gives us what we want as above. □

Note: The proof must go in this direction. For weight functions $w_i : A_i \to \mathbb{N}$, $\text{preim}_{w_i}(n)$ is finite but $\bigcup \text{preim}_{w_i}(n)$ is not guaranteed to be finite.

Lemma (*Product Lemma*)

Let A and B be sets and w_A and w_B be weight functions. Define $w_{A\times B} : A \times B \to \mathbb{N} : (a,b) \mapsto w_A(a) + w_B(b)$. Then, $\Phi_{A\times B}^{w_{A\times B}} = \Phi_A^{w_A} \cdot \Phi_B^{w_B}$.

Proof. Recall by definition $\Phi_{A\times B}^{w_{A\times B}} = \sum_{(a,b)\in A\times B} x^{w_{A\times B}((a,b))}$.

But this is $\sum_{(a,b)\in A\times B} x^{w_A(a)+w_B(b)} = \sum_{(a,b)\in A\times B} x^{w_A(a)}x^{w_B(b)} = \sum_{a\in A}\sum_{b\in B} x^{w_A(a)}x^{w_B(b)}$.

We split the sum to get $\left(\sum_{a\in A} x^{w_A(a)}\right)\left(\sum_{b\in B} x^{w_B(b)}\right) = \Phi_A^{w_A}(x) \cdot \Phi_B^{w_B}(x)$. □

Corollary (Finite Products). Define $w(a_1, \dots, a_k)$ on $A_1 \times \cdots \times A_k$ by $\sum_{i=1}^{k} w_{A_i}(a_i)$. Then,

$$\Phi^w_{A_1\cdots\times A_k}(x) = \prod_{i=1}^{k} \Phi_{A_i}^{w_{A_i}}(x)$$

Corollary (Cartesian Product Lemma). For set A with weight function w, $\Phi_{A^k}^{w_k}(x) = (\Phi_A^w(x))^k$ where $w_k((a_i)) = \sum w(a_i)$.

Definition (*strings*)

$A^* = \bigcup_{i=0}^{\infty} A^i$ where $A^0 = \{\varepsilon\}$ with the empty string $\varepsilon = \varnothing$. That is, A^* is the set of all strings (sequences) with entries from A.

Example 7.1. For $A = \mathbb{N} \setminus \{0\}$, then A^* is the set of all compositions. For $A = \{2, 4, 6, \dots\}$, then A^* is the set of all compositions with even parts.

Lecture 8 (09/23)

Lemma (*String Lemma; 2.14*)

Given A and w on A, define $w^* : A^* \to \mathbb{N} : (a_i) = \sum w(a_i)$ with $w^*(\varepsilon) = 0$. Also, there are no elements of A with weight 0. Then, $\Phi_{A^*}^{w^*}(x) = \frac{1}{1-\Phi_A^w(x)}$.

Proof. By definition, $A^* = A^0 \cup A^1 \cup \cdots$ and this is a disjoint union.

Then, since w^* is a weight function of A^* becaues A has no weight 0 elements, we apply the Infinite Sum Lemma. This gives $\Phi_{A^*}^{w^*}(x) = \sum \Phi_{A^i}^{w^*}(x)$.

By the Product Lemma, $\Phi_{A^i}^{w*}(x) = (\Phi_A^w(x))^i$. That is, $\Phi_{A^*}^{w^*}(x) = \sum(\Phi_A^w(x))^i$. Now, we can compose the geometric series $\frac{1}{1-x}$ with $\Phi_A^w(x)$ so long as $[x^0]\Phi_A^w(x) = 0$ which is true by supposition.

Therefore, we get $\Phi_{A^*}w^*(x) = \sum(\Phi_A^w(x))^i = \frac{1}{1-\Phi_A^w(x)}$. □

Example 8.1. Let $A = \mathbb{N}_{\geq 1}$. Then, A^* is the set of all compositions. Define $w = \mathrm{id}$, so A has no elements of weight 0 and we have w^* is a weight function. Namely, w^* gives the size of a composition.

By the String Lemma, $\Phi_{A^*}^{w*}(x) = \frac{1}{1-\Phi_A^w(x)}$. By inspection, $\Phi_A^w(x) = x + x^2 + \cdots$. By the geometric series, we have $x(1 + x + \cdots) = \frac{x}{1-x}$ so $\Phi_{A^*}^{w^*}(x) = \frac{1}{1-\frac{x}{1-x}} = \frac{1-x}{1-2x} = 1 + \frac{x}{1-2x}$ and we can write this as $1 + x + 2x^2 + 4x^3 + \cdots$ or

$$[x^n]\Phi_{A^*}^{w^*}(x) = \begin{cases} 1 & n = 0 \\ 2^{n-1} & n \geq 1 \end{cases}$$

Example 8.2. Let $A = 2\mathbb{N}_{\geq 1}$. Then, A^* is the set of all compositions with even parts and as above we can define $w = \text{id}$ and w^* is a weight funciton giving the size.

By inspection, $\Phi_A^w(x) = x^2 + x^4 + x^6 + \cdots = x^2(1 + (x^2) + (x^2)^2 + \cdots) = \frac{x^2}{1-x^2}$.

Then, $\Phi_{A^*}^{w^*}(x) = \frac{1}{1-\frac{x^2}{1-x^2}} = \frac{1-x^2}{1-2x^2} = 1 + \frac{x^2}{1-2x^2}$ and we can write

$$[x^n]\Phi_{A^*}^{w^*}(x) = \begin{cases} 1 & n = 0 \\ 2^{\frac{n}{2}-1} & n \text{ even} \\ 0 & n \text{ odd} \end{cases}$$

Example 8.3. $A = \{1, 2, 3, 4\}$, $w = \text{id}$, A^* is the set of all compositions with parts 1 to 4. Again, $\Phi_A^w(x) = x + x^2 + x^3 + x^4$ and $\Phi_{A^*}^{w^*} = \frac{1}{1-x-x^2-x^3-x^4}$.

Example 8.4. $A = \{1, 2\}$, $w = \text{id}$, $\Phi_A^w(x) = x + x^2$, $\Phi_{A^*}^{w^*}(x) = \frac{1}{1-x-x^2}$ which is the Fibonacci series.

Example 8.5. $A = \{6, 7, 8, \ldots, 100\}$, $\Phi_A^w(x) = x^6 + x^7 + \cdots + x^{100}$. By the Finite Geometric Series, this is $\frac{x^6-x^{101}}{1-x}$. Then, $\Phi_{A^*}^{w^*}(x) = \frac{1}{1-\frac{x^6-x^{101}}{1-x}} = \frac{1-x}{1-x^6+x^{101}}$.

Chapter 3

Binary Strings

Lecture 9 (09/26)

Definition (*binary string*)

A sequence of 0's and 1's. The *length* of a binary string is the total number of 0's and 1's. We notate the *empty string* by $\varepsilon = \varnothing$. Formally, it is an element of $\{0,1\}^*$ written without parentheses and commas.

Remark. Define a weight function for $\{0,1\}^*$ to get the length. Let $A = \{0,1\}$. Define $w(a) = 1$. Then if we apply Lemma 2.13, w^* is a weight function and by the String Lemma, $\Phi_{A^*}^{w^*}(x) = \frac{1}{1-\Phi_A^w(x)} = \frac{1}{1-2x}$.

Definition (*regular expression*)

A string of finite length that is, up to parentheses, any one of:

- ε, 0, and 1;
- $\mathsf{R} \smile \mathsf{S}$ for regular expressions R and S;
- RS for regular expressions R and S; or
- $\mathsf{R}*$ for regular expression R

Definition (*concatenation product*)

For binary strings $\alpha = \alpha_1 \ldots \alpha_n \in \{0,1\}^*$ and $\beta = \beta_1 \ldots \beta_n \in \{0,1\}^*$, define $\alpha\beta = \alpha_1 \ldots \alpha_n \beta_1 \ldots \beta_n$

For sets of binary strings $A, B \subseteq \{0,1\}^*$, define $AB = \{\alpha\beta : \alpha \in A, \beta \in B\}$.

Remark. The concatenation product acts like a flatten over the Cartesian product. That is, $((1,0),1) \neq (1,(0,1))$ but $(1,0)(1) = (1)(0,1)$. It follows that $|AB| \leq |A||B|$.

Definition (*rational language*)

$\mathcal{R} \subseteq \{0,1\}^*$ produced by a regular expression R:

- ε, 0, and 1 produce themselves;
- $\mathsf{R} \smile \mathsf{S}$ produces $\mathcal{R} \cup \mathcal{S}$;
- RS produces the concatenation product $\mathcal{R}\mathcal{S}$; and
- R^* produces $\bigcup_{k \geq 0} \mathcal{R}^k$ where $\mathcal{R}^k$ is the concatenation power.

Example 9.1. What languages do $(1)^*$, $(1 \smile 11)^*$, $(0 \smile 1)^*$, $1(11)^*$, $(01)^*$ produce?

Solution. $(1)^*$ produces the set of binary strings of only ones, i.e., $\{1\}^*$.

$(1 \smile 11)^*$ produces the same set.

$(0 \smile 1)^*$ produces $\{0,1\}^*$, the set of all binary strings.

$1(11)^*$ produces the set of all odd numbers of ones.

$(01)^*$ produces $\{\varepsilon, 01, 0101, 010101, \dots\}$. $\square$

Example 9.2. Not every set is a rational language. The set $\{\varepsilon, 01, 0011, 000111, 0^i1^i\}$ is not a rational language because it cannot be produced by a regular expression.

Lecture 10 (09/28; from Bradley)

Definition (*unambiguity*)

Regular expression R that produces every element in the corresponding rational language $\mathcal{R}$ exactly once.

Lemma

The following regular expressions are unambiguous:

- ε, 0, and 1
- $\mathsf{R} \smile \mathsf{S}$ given $\mathcal{R} \cap \mathcal{S} = \varnothing$
- RS given either (1) $|\mathcal{R}\mathcal{S}| = |\mathcal{R}||\mathcal{S}|$, (2) for all $t \in \mathcal{R}\mathcal{S}$, there is a unique $r \in \mathcal{R}$, $s \in \mathcal{S}$ such that $rs = t$, or (3) there does not exist $r_1, r_2 \in \mathcal{R}$, $s_1, s_2 \in \mathcal{S}$ such that $r_1s_1 = r_2s_2$.
- R^* given R unambiguous and either (1) all of ε, $\mathcal{R}$, and $\mathcal{R}^2$ are disjoint or (2) all of the $\mathcal{R}^i$ are unambiguous as products

Example 10.1. Consider the following:

- 1^* is unambiguous because ε, 1, 1^2, etc. are all disjoint meaning that 1^k is unambiguous for all k.
- $(1 \smile 11)^*$ is ambiguous because we can produce 11 as either $(11)^1$ or 1^2.
- $(0 \smile 1)^*$ produces all binary strings and is unambiguous. The union is clearly disjoint. The binary strings of lengths k are clearly disjoint.
- $(101)^*$ produces $\{\varepsilon, 101, 1011101, \dots\}$ and is unambiguous.
- $(1 \smile 10 \smile 01)^*$ is ambiguous because 101 is produced by $1^1(01)^1$ and $(10)^1 1^1$.

Definition

R leads to a rational function $R(x)$ according to its structure:

- ε leads to 1
- 0 leads to x
- 1 leads to 1
- $R \smile S$ leads to $R(x) + S(x)$
- RS leads to $R(x) \cdot S(x)$
- R^* leads to $\frac{1}{1-R(x)}$

Example 10.2. Consider the same regular expressions:

- 1^* leads to $\frac{1}{1-x}$
- $(1 \smile 11)^*$ leads to $\frac{1}{1-(x+x^2)} = \frac{1}{1-x-x^2}$
- $(0 \smile 1)^*$ leads to $\frac{1}{1-2x}$.
- $(101)^*$ leads to $\frac{1}{1-x^3}$

- $(1 \smile 10 \smile 01)^*$ leads to $\frac{1}{1-x-2x^2}$
- $0^*(11^*00^*)^*1^*$ leads to $\frac{1}{1-x}\frac{1}{1-\frac{x^2}{(1-x)^2}}\frac{1}{1-x} = \frac{1}{1-2x}$

Lemma

Let R be a regular expression, $\mathcal{R}$ its rational language, $R(x)$ the rational function it leads to, and w the length weight function on binary strings.
If R is unambiguous, then $\Phi^w_{\mathcal{R}}(x) = R(x)$.

Proof. Proceed by structural induction on the above unambiguity lemma:

Notice that $\Phi^w_{\{e\}}(x) = 1$, $\Phi^w_{\{0\}}(x) = x$, and $\Phi^w_{\{1\}}(x) = x$.

If $\mathsf{R} = \mathsf{S} \smile \mathsf{T}$ and is unambiguous, then R produces $\mathcal{S} \cup \mathcal{T}$ and

$$\Phi^w_{\mathcal{S}\cup\mathcal{T}}(x) = \Phi^w_{\mathcal{S}}(x) + \Phi^w_{\mathcal{T}}(x) = S(x) + T(x) = R(x)$$

by the Sum Lemma.

Likewise by the Product Lemma, if $\mathsf{R} = \mathsf{ST}$ we can write

$$\Phi^w_{\mathcal{S}\mathcal{T}}(x) = \Phi^w_{\mathcal{S}\times\mathcal{T}}(x) = \Phi^w_{\mathcal{S}}(x) \cdot \Phi^w_{\mathcal{T}}(x) = S(x) \cdot T(x) = R(x)$$

because $\mathcal{S}\mathcal{T} = \mathcal{S} \times \mathcal{T}$ by unambiguity.

The case for $\mathsf{R} = \mathsf{S}^*$ goes similarly by the Infinite Sum Lemma and Product Lemma. □

Be careful to distinguish between:

- A regular expression R
- A rational language $\mathcal{R} \subseteq \{0,1\}^*$ it *produces*
- A rational function $R(x) \in \mathbb{Z}[[x]]$ it *leads to*, equal to $\Phi_{\mathcal{R}}(x)$ when R is unambiguous

Lecture 11 (09/30; from Bradley)

Definition (*block*)

A maximal subsequence of a binary string with the same digit.

Lemma (*Block Decompositions*)

The set of all binary strings is unambiguously produced by $0^*(11^*00^*)^*1^*$ and $1^*(00^*11^*)^*0^*$.

Proof. WLOG consider the second regular expression. We decompose every binary string after each block of 0s.

This means each string is of the form (a (possibly empty) initial block of 0s, first pair of blocks of 1s, second pair of 0s, ..., last pair, (possibly empty) terminal block of 1s). Moreover, first/last pair may not exist.

This decomposition is unique and we can express it as a regular expression of the form $\mathsf{I}(\mathsf{M})^*\mathsf{T}$ where

- I is a regular expression for the initial (possibly empty) block of 0s
- M is a regular expression for a middle pair of blocks, non-empty block of 1s followed by non-empty block of 0s

- T is a regular expression for the terminal (possibly empty) block of 1s

Then, we can unambiguously write $\mathsf{I} = 0^*$, $\mathsf{M} = (11^*00^*)^*$, and $\mathsf{T} = 1^*$. Using the extra 1/0 ensures each block is non-empty. □

Example 11.1. Write an unambiguous regular expression for "the set of binary strings where each block of 0s has length at least 2 and each block of 1s has even length".

Solution. Follow the $\mathsf{I}(\mathsf{M})^*\mathsf{T}$ pattern beginning with a block of 0s.

To get either no 0s or at least two 0s, set $\mathsf{I} = \varepsilon \smile (000^*)$.

To get either no 1s or an even number of 1s, set $\mathsf{T} = (11)^*$.

Combine these ideas to get $\mathsf{M} = 11(11)^*000^*$.

Altogether, write $(\varepsilon \smile (000^*))(11(11)^*000^*)^*(11)^*$. □

Example 11.2. Write an unambiguous regular expression for "the set of all binary strings where each block of 0s has length at least 5 and congruent to 2 (mod 3) and each block of 1s has length at least 2 and at most 8".

Solution. Follow the $\mathsf{I}(\mathsf{M})^*\mathsf{T}$ pattern beginning with a block of 0s.

To get either no 0s or $5 + 3k$ 0s, set $\mathsf{I} = \varepsilon \smile (00000(000)^*)$.

To get either no 1s or between two and eight 1s, set $\mathsf{T} = (11 \smile 111 \smile \cdots \smile 1^8 \smile \varepsilon)$.

Combine to get $\mathsf{M} = (11 \smile 111 \smile \cdots \smile 1^8)(00000(000)^*)$. Altogether,

$$(\varepsilon \smile (00000(000)^*))((11 \smile 111 \smile \cdots \smile 1^8)(00000(000)^*))^*(11 \smile 111 \smile \cdots \smile 1^8 \smile \varepsilon)$$

□

Example 11.3. Same as example 11.2 with the additional restriction that the string is non-empty and starts with 0.

Solution. $(0^5(000)^*)((1^2 \smile 1^3 \smile \cdots \smile 1^8)(0^5(000)^*))^*(1^2 \smile 1^3 \smile \cdots \smile 1^8 \smile \varepsilon)$.

If we wanted it to start with 1 instead, decompose the blocks beginning with 1s to get a similar answer: $(1^2 \smile 1^3 \smile \cdots \smile 1^8)((0^5(000)^*)(1^2 \smile 1^3 \smile \cdots \smile 1^8))^*(0^5(000)^* \smile \varepsilon)$. □

Example 11.4. Starts with 0 and at least 2 blocks

Solution. $(0^5(000)^*)(1^2 \smile \cdots \smile 1^8)((0^5(000)^*)(1^2 \smile \cdots \smile 1^8))^*(0^5(000)^* \smile \varepsilon)$. □

Example 11.5. Starts with 0 and ends with 0

Solution. $(0^5(000)^*)((1^2 \smile 1^3 \smile \cdots \smile 1^8)(0^5(000)^*))^*$. □

Lecture 12 (10/03)

Instead of decomposing after every block of 1s, we could instead decompose after every 1. This leads to prefix decompositions: the regular expression $\mathsf{M}^*\mathsf{T}$ where $\mathsf{M} = 0^*1$ and $\mathsf{T} = 0^*$ unambiguously produces the set of all binary strings.

As a sanity check, notice that $(M^*T)(x) = \frac{1}{1-\frac{1}{1-x}x}\frac{1}{1-x} = \frac{1}{1-2x}$ which we expect as the generating series for all binary strings.

Example 12.1. Write an unambiguous regular expression for binary strings without k consecutive zeroes.

Solution. $((\varepsilon \smile 0^1 \smile \cdots \smile 0^{k-1})1)^*(\varepsilon \smile 0^1 \smile \cdots \smile 0^{k-1})$.

This leads to $\frac{1}{1-(1+\cdots+x^{k-1})x} \cdot (1+\cdots+x^{k-1}) = \frac{1+\cdots+x^{k-1}}{1-x-\cdots-x^k} = \frac{1-x^k}{1-2x+x^{k+1}}$ □

Similarly, decompose before every 1. This leads to postfix decompositions: the regular expression $\mathsf{I}\mathsf{M}^*$ where $\mathsf{I} = 0^*$ and $\mathsf{M} = 10^*$ unambiguously produces the set of all binary strings.

Definition (*recursive expression*)

A regular expression R which may contain itself. The rational function $R(x)$ that R leads to is identical to if it were a regular expression except R leads to $R(x)$.

Example 12.2. Interpet $\mathsf{S} = \varepsilon \smile (0 \smile 1)\mathsf{S}$.

Solution. Clearly, ε is in $\mathcal{S}$. Also, $0\varepsilon = 0$ and $1\varepsilon = 1$ are in $\mathcal{S}$. Onward, by induction, $\mathcal{S}$ contains all binary strings.

The rational function $S(x) = 1 + (x+x)S(x) = 1 + 2xS(x) \implies S(x) = \frac{1}{1-2x}$ which matches the generating series for binary strings. □

Example 12.3. Interpret $\mathsf{S} = \varepsilon \smile 1\mathsf{S}$.

Solution. This produces the set of all binary strings with no 0s.

It leads to $S(x) = 1 + xS(x) \implies S(x) = \frac{1}{1-x}$. □

Example 12.4. Write the prefix decomposition as a recursive expression.

Solution. $\mathsf{S} = 0^* \smile (0^*1)\mathsf{S}$ □

Example 12.5. Write example 12.1 as a recursive expression.

Solution. $\mathsf{S} = (\varepsilon \smile 0 \smile \cdots \smile 0^{k-1}) \smile ((\varepsilon \smile 0 \smile \cdots \smile 0^{k-1})1)\mathsf{S}$.

This leads to $S(x) = (1+\cdots+x^{k-1}) + (1+\cdots+x^{k-1})S(x) \implies S(x) = \frac{1+\cdots+x^{k-1}}{1-(1+x+\cdots+x^{k-1})x}$ as in example 12.1. □

Example 12.6. Write example 9.2 as a recursive expression.

Solution. $\mathsf{S} = \varepsilon \smile 0\mathsf{S}1$. Recall that this is not possible with ordinary regular expressions.

Then, the generating series $S(x) = 1 + xS(x)x = 1 + x^2S(x) \implies S(x) = \frac{1}{1-x^2}$. □

Recursion allows us to solve a wider range of problems.

Definition (*containment*)

A binary string σ contains κ if there exist binary strings α and β such that $\sigma = \alpha\kappa\beta$.

We want to solve the general problem: given κ a fixed binary string and $\mathcal{S}_\kappa$ the set of all binary strings that do not contain κ, determine $\Phi_{\mathcal{S}_\kappa}(x)$.

We just did this for $\kappa = 0^k$ in examples 12.1 and 12.5, but what about arbitrary κ?

Lecture 13 (10/05)

Definition (*avoidance*)

A binary string α excludes a binary string κ if α does not contain κ.

We are solving the general problem: given a binary string κ, let A_κ be the set of binary strings avoiding κ. Determine $\Phi_{A_\kappa}(x)$.

Theorem *(3.26)*

Let κ be a fixed binary string, $\mathcal{A} = \mathcal{A}_\kappa$ be the set of binary strings avoiding κ, and $A(x) = \Phi_{\mathcal{A}_\kappa}(x)$. Then,

$$A(x) = \frac{1 + C(x)}{(1 - 2x)(1 + C(x)) + x^{\ell(\kappa)}}$$

where $C(x) = \Phi_{\mathcal{C}}(x)$ where $\mathcal{C}$ is the set of non-empty proper suffixes γ of κ such that there exists a non-empty proper prefix η such that $\kappa\gamma = \eta\kappa$.

Example 13.1. $\kappa = 0^k$. Then, $\mathcal{C} = \{0, 0^2, \dots, 0^{k-1}\}$ because $0^k 0^i = 0^i 0^k$.

Example 13.2. $\kappa = 11011$. Then, $\mathcal{C} = \{011, 1011\}$ because we can write $(11011)(011) = (110)(11011)$ and $(11011)(1011) = (1101)(11011)$ but not with 1 or 11.

Proof. Let $\mathcal{B}$ be the set of all binary strings with exactly one occurence of κ and that occurence is at the end. Let $B(x) = \Phi_{\mathcal{B}}(x)$.

Claim. The following are unambiguous recursive expressions relating $\mathcal{A}$ and $\mathcal{B}$:

$$\mathsf{A} \smile \mathsf{B} = \varepsilon \smile \mathsf{A}(0 \smile 1) \qquad\qquad \mathsf{A}\kappa = \mathsf{B} \smile \underset{\gamma\in\mathcal{C}}{\smile} \mathsf{B}\gamma$$

For (1): We know $\varepsilon \in \mathcal{A}$ and $\mathsf{A}(0 \smile 1)$ generates the strings of length at least 1 that are either in $\mathcal{A}$ (by adding a bit that does not complete κ) or in $\mathcal{B}$ (by adding a bit that does). In the other direction, notice that deleting a bit from $\mathcal{A}$ or $\mathcal{B}$ is in $\mathcal{A}$.

For (2): If $\sigma \in \mathcal{A}\kappa$, then σ has an occurence of κ at the end.

Let σ' be the substring of σ with the same start and ending at the first occurence of κ. This exists since there exists at least one occurence of κ.

Thus, $\sigma' \in \mathcal{B}$ and $\sigma = \sigma'\gamma = \sigma''\kappa$ and hence $\kappa\gamma = \eta\kappa$ and $\gamma \in \mathcal{C}$ where γ is the suffix of κ of length $w(\sigma) = w(\sigma')$.

Converting the expressions to equations gives

$$A(x) + B(x) = 1 + A(x)2x \qquad\qquad \begin{aligned} A(x)x^{w(\kappa)} &= B(x)\left(1 + \sum_{\gamma\in\mathcal{C}} x^w(\gamma)\right) \\ &= B(x)(1 + C(x)) \end{aligned}$$

and solving gives us $A(x) = \frac{1+C(x)}{(1+2x)(1+C(x))+x^{w(\kappa)}}$ by eliminating $B(x)$. $\square$

Chapter 4

Recurrence Relations