Chapter 4: Generating Functions

In this chapter, we introduce Generating Functions, a fundamental problem-solving tool in combinatorics. We treat these functions as **formal power series**. This means we are interested in the algebraic properties of their coefficients and the structural information they carry, rather than the convergence of the series for numerical values of x.

We will organize our study into three main parts:

1. **Ordinary Generating Functions (OGFs):** Used primarily for selection problems where order does not matter.
2. **Exponential Generating Functions (EGFs):** Used primarily for arrangement problems where order matters (labeled structures).
3. **Multivariate Generating Functions:** Used for problems involving multiple parameters.

1. Preliminaries: Manipulation of Sums

Summations are the engine room of discrete mathematics, particularly when dealing with Generating Functions and recurrence relations. Before we define generating functions, we must master the tools to manipulate these sums. These properties allow us to simplify expressions, align indices, and extract coefficients effectively.

- **Linearity:** The summation operator is linear. We can split sums and factor out constants.
$$\sum_{k=m}^{n}(c \cdot a_k + d \cdot b_k) = c\sum_{k=m}^{n} a_k + d\sum_{k=m}^{n} b_k.$$
- **Splitting (Associativity):** We can break a sum into two contiguous parts. For any integer p such that $m \leq p < n$:
$$\sum_{k=m}^{n} a_k = \sum_{k=m}^{p} a_k + \sum_{k=p+1}^{n} a_k.$$
- **Index Shifting (Substitution):** This is the discrete equivalent of "u-substitution" in integrals. To shift the index by r, we replace k with $j-r$:
$$\sum_{k=m}^{n} a_k = \sum_{j=m+r}^{n+r} a_{j-r}.$$
Usage: This is crucial in Generating Functions for aligning powers of x.
- **Reversing Order:** We can sum from the end to the beginning by substituting $j = n-k$.
$$\sum_{k=0}^{n} a_k = \sum_{j=0}^{n} a_{n-j}.$$
Usage: This property is often used to recognize convolutions.
- **Double Sum Swapping (Fubini's Principle):** For finite sums with independent bounds, the order of summation is interchangeable:
$$\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij} = \sum_{j=1}^{m}\sum_{i=1}^{n} a_{ij}$$
- **Separable Variables (Product Rule):** If the term a_{ij} can be factored into $f(i) \cdot g(j)$, the double sum simplifies to the product of two single sums:
$$\sum_{i=1}^{n}\sum_{j=1}^{m} f(i)g(j) = \left(\sum_{i=1}^{n} f(i)\right)\left(\sum_{j=1}^{m} g(j)\right)$$
- **Dependent Bounds:** If the inner limit depends on the outer index ($1 \leq j \leq i \leq n$), the indices swap as follows:
$$\sum_{i=1}^{n}\sum_{j=1}^{i} a_{ij} = \sum_{j=1}^{n}\sum_{i=j}^{n} a_{ij}$$
- **General Double Sum Swapping:** For any finite index sets I and J, if the summation limits

are independent:

$$\sum_{i\in I}\left(\sum_{j\in J} a_{ij}\right) = \sum_{(i,j)\in I\times J} a_{ij} = \sum_{j\in J}\left(\sum_{i\in I} a_{ij}\right)$$

- **Separable Variables (Product Rule):** If a_{ij} can be factored into $f(i)g(j)$, the sum over the Cartesian product $I \times J$ simplifies to:

$$\sum_{i\in I}\sum_{j\in J} f(i)g(j) = \left(\sum_{i\in I} f(i)\right)\left(\sum_{j\in J} g(j)\right)$$

- **Dependent Bounds (The Subset Rule):** If the index j depends on i (i.e., $(i,j) \in S \subseteq I \times J$), we sum over the relation S:

$$\sum_{i\in I}\sum_{j\in J(i)} a_{ij} = \sum_{(i,j)\in S} a_{ij} = \sum_{j\in J}\sum_{i\in I(j)} a_{ij}$$

where $J(i) = \{j : (i,j) \in S\}$ and $I(j) = \{i : (i,j) \in S\}$.

- **The Telescoping Sum:** If a term can be written as a difference $a_k = b_{k+1} - b_k$, the intermediate terms cancel out.

$$\sum_{k=m}^{n} (b_{k+1} - b_k) = b_{n+1} - b_m.$$

These properties will be used extensively in the following sections to transform recurrence relations into algebraic equations.

2. Ordinary Generating Functions

We begin with the most common type of generating function, used to encode sequences where the position in the sequence corresponds to the power of x.

2.1 Definition and Basic Examples

Definition 2.1 (Ordinary Generating Function (OGF)).
The **Ordinary Generating Function** (OGF) for an infinite sequence $\{a_n\}_{n=0}^{\infty} = (a_0, a_1, a_2, \dots)$ is the formal power series:

$$G(x) = \sum_{n=0}^{\infty} a_n x^n = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \dots \tag{1}$$

In this series, the coefficient a_n is the n-th term of the sequence. This series is often represented by a *compact expression* (or closed form). Expanding this expression generates a power series having the terms of the sequence as coefficients.

Example (Constant Sequence).
Consider the sequence $(1, 1, 1, 1, \dots)$, where $a_n = 1$ for all $n \geq 0$. Its generating function is the geometric series:

$$f(x) = \sum_{n\geq 0} x^n = 1 + x + x^2 + \dots$$

In closed form (formally), this is:

$$f(x) = \frac{1}{1-x}.$$

Example (Alternating Sequence).
Consider the sequence $(1, -1, 1, -1, \dots)$. The OGF is:

$$A(x) = 1 - x + x^2 - x^3 + \dots = \sum_{n=0}^{\infty} (-1)^n x^n.$$

Using the geometric series formula with $-x$, we get:

$$A(x) = \frac{1}{1+x}.$$

2.2 Operations on Ordinary Generating Functions

To solve problems effectively, we need to know how manipulating a sequence affects its generating function, and vice versa. Table 3 summarizes these fundamental operations.

Let $f(x)$ and $g(x)$ be the OGFs associated with sequences $(a_n)_{n\geq 0}$ and $(b_n)_{n\geq 0}$ respectively.

Rule	Sequence $(c_n)_{n\geq 0}$	OGF $h(x)$
Scaling	$\lambda a_0, \lambda a_1, \lambda a_2, \dots$	$h(x) = \lambda f(x)$
Addition	$(a_0 + b_0), (a_1 + b_1), \dots$	$h(x) = f(x) + g(x)$
k-right-shift	$\underbrace{0, 0, \dots, 0}_{k \ zeroes}, a_0, a_1, \dots$	$h(x) = x^k f(x)$
k-left-shift	$a_k, a_{k+1}, a_{k+2}, \dots$	$h(x) = \frac{f(x) - \sum_{i=0}^{k-1} a_i x^i}{x^k}$
Differentiation	$a_1, 2a_2, 3a_3, \dots, na_n, \dots$	$h(x) = f'(x)$
Integration	$0, a_0, \frac{a_1}{2}, \frac{a_2}{3}, \dots$	$h(x) = \int_0^x f(z)dz$
Difference	$a_0, (a_1 - a_0), (a_2 - a_1), \dots$	$h(x) = (1 - x)f(x)$
Product (Cauchy)	$a_0 b_0, a_1 b_0 + a_0 b_1, \dots$	$h(x) = f(x)g(x)$

Table 3: Operations on OGFs.

Definition 2.2 (Addition and Scaling).
Based on the table above, if $A(x) \leftrightarrow (a_n)$ and $B(x) \leftrightarrow (b_n)$, then:

$$A(x) + B(x) \leftrightarrow (a_n + b_n)$$
$$cA(x) \leftrightarrow (ca_n) \quad \text{for any constant } c.$$

One of the most powerful operations is the product of two generating functions, which corresponds to the convolution of their sequences.

Theorem 2.3 (Convolution / Product Rule).
Let $A(x) = \sum a_n x^n$ and $B(x) = \sum b_n x^n$. The product $C(x) = A(x)B(x)$ is the generating function for the sequence (c_n), where:

$$c_n = \sum_{k=0}^{n} a_k b_{n-k}.$$

The sequence (c_n) is called the **convolution** of (a_n) and (b_n).

Example (Square of the Geometric Series).
Let $A(x) = \frac{1}{1-x}$, which corresponds to $a_n = 1$. Consider $A(x)^2 = \left(\frac{1}{1-x}\right)^2$. By the convolution

theorem, the coefficient c_n is:

$$c_n = \sum_{k=0}^{n} a_k a_{n-k} = \sum_{k=0}^{n} 1 \cdot 1 = n + 1.$$

Thus, $\frac{1}{(1-x)^2} = \sum_{n=0}^{\infty}(n+1)x^n$.

2.3 Standard OGF Sequences

Using the operations defined above, we can derive the OGFs for other common sequences, such as linear sequences.

Theorem 2.4 (OGF for Linear Sequences).
Let $a_n = n$. Then:

$$\sum_{n=1}^{\infty} nx^n = \frac{x}{(1-x)^2}.$$

Theorem 2.5 (Generating Function of Fibonacci Numbers).
Let $F_0 = 0$, $F_1 = 1$, and $F_n = F_{n-1} + F_{n-2}$ for $n \geq 2$. The generating function of the Fibonacci sequence is:

$$F(x) = \sum_{n=0}^{\infty} F_n x^n = \frac{x}{1 - x - x^2}.$$

Exercise.
Prove that the generating function for the sequence of triangular numbers $T_n = \frac{n(n+1)}{2}$ is $\frac{x}{(1-x)^3}$.

Solution.
We know from Theorem 4.2 that $\sum_{n=1}^{\infty} nx^n = \frac{x}{(1-x)^2}$. We can rewrite the general term as:

$$\sum_{n=1}^{\infty} T_n x^n = \frac{1}{2}\sum_{n=1}^{\infty} n(n+1)x^n = \frac{1}{2}\sum_{n=1}^{\infty}(n^2+n)x^n.$$

To find the sum for n^2, we apply the differentiation operator ($x\frac{d}{dx}$) to the known series for n:

$$\sum_{n=1}^{\infty} n^2 x^n = x\frac{d}{dx}\left(\frac{x}{(1-x)^2}\right) = \frac{x(1+x)}{(1-x)^3}.$$

Substituting this back into the expression for T_n:

$$\sum_{n=1}^{\infty} T_n x^n = \frac{1}{2}\left(\frac{x(1+x)}{(1-x)^3} + \frac{x}{(1-x)^2}\right) = \frac{1}{2} \cdot \frac{x(1+x) + x(1-x)}{(1-x)^3} = \frac{x}{(1-x)^3}.$$

3. Exponential Generating Functions

While OGFs are excellent for unlabelled objects, combinatorial problems involving labeled structures (like permutations) are often better handled using Exponential Generating Functions.

3.1 Definition and Basic Examples

Definition 3.1 (Exponential Generating Function (EGF)).
The **Exponential Generating Function** (EGF) for a sequence $(a_n)_{n\geq 0}$ is the formal power series:

$$E(x) = \sum_{n=0}^{\infty} a_n \frac{x^n}{n!} = a_0 + a_1\frac{x}{1!} + a_2\frac{x^2}{2!} + \ldots \quad (2)$$

Note the division by $n!$ in the n-th term.

Example (Constant Sequence and Powers).

The EGF of the sequence $a_n = m^n$ for all $n \geq 0$, where m is a constant, is:

$$F(x) = \sum_{n\geq 0} m^n \frac{x^n}{n!} = 1 + \frac{mx}{1!} + \frac{(mx)^2}{2!} + \cdots + \frac{(mx)^n}{n!} + \cdots = e^{mx}.$$

Specifically, if $m = 1$ (the constant sequence $a_n = 1$), the EGF is e^x.

Example (Permutations)**.**
Consider the sequence of factorials, $a_n = n!$ (the number of permutations of n items).

- The OGF would be $\sum n!x^n$, which does not have a simple closed form.
- The EGF is much simpler:
$$F(x) = \sum_{n=0}^{\infty} n! \frac{x^n}{n!} = \sum_{n=0}^{\infty} x^n = \frac{1}{1-x}.$$

3.2 Operations on Exponential Generating Functions

Just as with OGFs, we have a set of operations for EGFs. Let $F(x)$ and $G(x)$ be the EGFs associated with sequences $(a_n)_{n\geq 0}$ and $(b_n)_{n\geq 0}$.

Rule	Sequence $(c_n)_{n\geq 0}$	EGF $H(x)$
Scaling	$\lambda a_0, \lambda a_1, \ldots$	$H(x) = \lambda F(x)$
Addition	$(a_0 + b_0), (a_1 + b_1), \ldots$	$H(x) = F(x) + G(x)$
k-right-shift	$\underbrace{0, \ldots, 0}_{k}, a_0, \ldots$	$H(x) = \underbrace{\int \cdots \int}_{k} F(x)dx$
k-left-shift	$a_k, a_{k+1}, \ldots$	$H(x) = \frac{d^k}{dx^k} F(x)$
Index multiply	$0, a_1, 2a_2, \ldots$	$H(x) = xF'(x)$
Index divide	$a_1, \frac{a_2}{2}, \ldots$	$H(x) = \frac{F(x)-F(0)}{x}$
Difference	$(a_1 - a_0), (a_2 - a_1), \ldots$	$H(x) = F'(x) - F(x)$
Binomial sum	$\sum_{k=0}^{n} \binom{n}{k} a_k$	$H(x) = e^x F(x)$
Binomial convolution	$\sum_{k=0}^{n} \binom{n}{k} a_k b_{n-k}$	$H(x) = F(x)G(x)$

Table 4: Operations on EGFs.

The product rule for EGFs differs from OGFs because it automatically handles the binomial coefficients arising from choosing labels.

Theorem 3.2 (Product of EGFs / Binomial Convolution)**.**
If $A(x)$ and $B(x)$ are the EGFs for (a_n) and (b_n), then $A(x)B(x)$ is the EGF for the sequence (c_n)

defined by:

$$c_n = \sum_{k=0}^{n} \binom{n}{k} a_k b_{n-k}.$$

This accounts for the ways to distribute labels between two structures.

3.3 Standard EGF Sequences

We can look at the linear sequence through the lens of EGFs.

Theorem 3.3 (EGF for Linear Sequences).
Let $a_n = n$. Then:

$$\sum_{n=0}^{\infty} \frac{nx^n}{n!} = xe^x.$$

4. Multivariate Generating Functions

For combinatorial objects that have multiple parameters (e.g., a binary string with length n and k ones), we use functions with multiple variables.

Definition 4.1 (Bivariate Generating Function).
Let $(a_{n,k})_{n,k\geq 0}$ be a doubly indexed sequence. The bivariate generating function is:

$$A(x, y) = \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} a_{n,k} x^n y^k.$$

Definition 4.2 (Mixed Generating Function).
Given a double sequence $(a_{n,k})_{n,k\geq 0}$, its **mixed ordinary-exponential generating function** is:

$$A(x, y) = \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} a_{n,k} x^n \frac{y^k}{k!}.$$

Example (Binomial Coefficients).
Let $a_{n,k} = \binom{n}{k}$. The bivariate generating function is:

$$A(x, y) = \sum_{n=0}^{\infty} \left(\sum_{k=0}^{n} \binom{n}{k} y^k \right) x^n = \sum_{n=0}^{\infty} (1 + y)^n x^n = \frac{1}{1 - x(1 + y)}.$$