Contents

1

Combinatorics

1.1   Basic Counting Principles

An important motivation to study combinatorics is to count the **number of ways** in which an event may occur. Intuitively, we have two approaches to count.

The first approach is to categorise the event into **non-overlapping cases**. This means that we break an event into mutually exclusive sub-events, after which we can count the number of ways for each sub-event to occur. The agregate of these counts is the total number of ways for the original event to occur.

Those familiar with basic set theory may consider E to be the set containing all distinct ways for an event to occur. By breaking up the event, we essentially establish a **partition** of E, so that the sum of cardinalities of all the elements in that partition equals the cardinality of E.

This motivates us to write the following principle using set notations.

Theorem 1.1.1 ▶ Addition Principle (AP)

Let $k \in \mathbb{N}^+$ and let $A_1, A_2, \cdots, A_k$ be k finite sets which are pairwise disjoint, i.e. $A_i \cap A_j = \varnothing$ whenever $i \neq j$, then

$$\left| \bigcup_{i=1}^{k} A_i \right| = \sum_{i=1}^{k} |A_i|.$$

Proof. The case where $k = 1$ is trivial.
Suppose that when $k = n$, we have

$$\left| \bigcup_{i=1}^{n} A_i \right| = \sum_{i=1}^{n} |A_i|$$

for any n finite sets which are pairwise disjoint. Let A_{n+1} be an arbitrary finite set

which is disjoint with any of the A_i's from the n sets. So we have:

$$
\begin{aligned}
\left| \bigcup_{i=1}^{n+1} A_i \right| &= \left| \left(\bigcup_{i=1}^{n} A_i \right) \cup A_{n+1} \right| \\
&= \left| \bigcup_{i=1}^{n} A_i \right| + |A_{n+1}| - \left| \left(\bigcup_{i=1}^{n} A_i \right) \cap A_{n+1} \right| \\
&= \left(\sum_{i=1}^{n} |A_i| \right) + |A_{n+1}| - |\varnothing| \\
&= \sum_{i=1}^{n+1} |A_i|.
\end{aligned}
$$

Therefore, the original statement holds for all $k \in \mathbb{N}^+$. $\square$

> *Remark.* In more casual language, this means that if an event E_k has n_k distinct ways to occur, then there is $\sum_{i=1}^{k} n_k$ ways for at least one of the events $E_1, E_2, \cdots, E_k$ to occur, provided that E_i and E_j can never occur concurrently whenever $i \neq j$.

Given an event E, the other approach to count the number of ways for it to occur is to break E up internally into **non-overlapping stages**.

With set notations, we can write the i-th stage for E to occur as e_i, and so a way for E to occur can be represented by an ordered tuple $(e_1, e_2, \cdots, e_k)$, where k is the total number of stages to undergo for E to occur.

Let E_i denote the set of all distinct ways to undergo the i-th stage of E, then it is easy to see that E is just the **Cartesian product** of all the E_i's. Hence, we derive the following principle:

Theorem 1.1.2 ▶ Multiplication Principle (MP)

Let $k \in \mathbb{N}^+$ and let $A_1, A_2, \cdots, A_k$ be k pairwise disjoint finite sets, then

$$
\left| \prod_{i=1}^{k} A_i \right| = \prod_{i=1}^{k} |A_i|.
$$

Proof. The case where $k = 1$ is trivial.
Suppose that when $k = n$, we have

$$
\left| \prod_{i=1}^{n} A_i \right| = \prod_{i=1}^{n} |A_i|
$$

for any n finite sets which are pairwise disjoint. Let A_{n+1} be an arbitrary finite set which is disjoint with any of the A_i's from the n sets. Take $a_i, a_j \in A_{n+1}$. Note that for all $\boldsymbol{a} \in \prod_{i=1}^{n} A_i, (\boldsymbol{a}, a_i) \neq (\boldsymbol{a}, a_j)$ whenever $a_i \neq a_j$. This means that

$$
\begin{aligned}
\left| \prod_{i=1}^{n+1} A_i \right| &= \left| \prod_{i=1}^{n} A_i \times A_{n+1} \right| \\
&= \left| \prod_{i=1}^{n} A_i \right| |A_{n+1}| \\
&= \left(\prod_{i=1}^{n} |A_i| \right) |A_{n+1}| \\
&= \prod_{i=1}^{n+1} |A_i|
\end{aligned}
$$

Therefore, the original statement holds for all $k \in \mathbb{N}^+$. $\qquad\square$

Remark. In more casual language, this means that if an event E requires k stages to be undergone before it occurs and the i-th stage has n_i ways to complete, then there is $\prod_{i=1}^{k} n_k$ ways for E to occur, provided that no two different stages complete concurrently.

Often times, it is not straight-forward to count directly due to the presence of restrictions. We shall consider the following:

Let E be the set of all possible ways for an event to occur. Let p be some predicate representing some restriction and let $E(p)$ denote the set of all possible ways for the event to occur while p holds. Note that:

$$
E(p) \cup E(\neg p) = E \quad \text{and} \quad E(p) \cap E(\neg p) = \varnothing,
$$

i.e. $\{E(p), E(\neg p)\}$ is a partition of E. Therefore, to count the number of ways for the event to occur while p holds, it suffices to compute $E(\neg p)$, i.e. find the number of ways for the event to occur while p does not hold.

Theorem 1.1.3 ▸ Principle of Complementation

Let U be a set and let $E \subseteq U$, then

$$
|E| = |U| - |U - E|.
$$

Remark. It may also help to think the Principle of Complementation as an inverse of the Addition Principle, where

$$E = \bigcup_{i=1}^{n} E_i$$

is the total number of ways for an event to occur and E_p is the number of ways for the event to occur with restriction p.

In some cases, it is difficult to count the objects directly. However, note that the events for which we are counting are just sets, so we can treat "number of occurrences" of an event as the **cardinality** of a set. Therefore, we have the following principles:

Theorem 1.1.4 ▶ Injection Principle

Let A, B be finite sets. If there exists an injection

$$f : A \hookrightarrow B,$$

then $|A| \leq |B|$.

Theorem 1.1.5 ▶ Bijection Principle

Let A, B be finite sets. If there exists a bijection

$$f : A \to B,$$

then $|A| = |B|$.

We will finish this section with two interesting problems.

Example 1.1.6 ▶ No Consecutive

Let $X = \{x \in \mathbb{N}^+ : x \leq n\}$. Find the number of r-combinations of X such that $|x_i - x_j| \geq 2$ whenever $i \neq j$.

Solution. Note that for each of the r-combinations of X, we can re-write it into a strictly increasing sequence $\{x_i\}_{i=1}^{r}$ with $x_i \in X$ and $x_{i+1} - x_i \geq 2$. Let the set of all such sequences be S_X. Consider the function

$$f : S_X \to Y$$

where $f\left(\{x_i\}_{i=1}^{r}\right) = \{x_i - i + 1\}_{i=1}^{r}$ and Y is the set of all strictly increasing sequences whose terms are integers bounded between 1 and $n - r + 1$ inclusive.

Take two different sequences $s_1, s_2 \in S_X$. Let the k-th terms of s_1 and s_2 be k_1 and k_2 respectively such that $k_1 \neq k_2$. Let $t_1 = f(s_1)$ and $t_2 = f(s_2)$, whose k-th terms are h_1 and h_2 respectively. Note that $h_1 = k_1 - k + 1 \neq k_2 - k + 1 = h_2$, so $t_1 \neq t_2$, which means f is injective.

Take an arbitrary $\{y_i\}_{i=1}^r \in Y$ and consider the sequence $\{y_i + i - 1\}_{i=1}^r$. Note that $1 \leq y_i \leq n - r + 1$, so $1 \leq y_i + i - 1 \leq n$ for $i = 1, 2, \cdots, r$, which means $\{y_i + i - 1\}_{i=1}^r \in S_X$. This means that for all $y \in Y$, there exists some $s \in S_X$ such that $f(s) = y$, so f is surjective.

Therefore, f is a bijection and so

$$|S_X| = |Y| = C_r^{n-r+1}.$$

Example 1.1.7 ▶ Top Secret

Six scientists are working on a secret project. They wish to lock up the documents in a cabinet so that the cabinet can be opened when and only when three or more of the scientists are present. Suppose that each lock has one and only one key with the locks being pairwise distinct.

1. What is the smallest number of locks needed?
2. What is the smallest number of keys that each scientist needs to carry?

Solution. Let $S = \{s_1, s_2, s_3, s_4, s_5, s_6\}$ be the set of scientists and L be the set of locks. Define T to be the set of all subsets of S with 2 elements, then for any $P \in T$, P is a pair of distinct scientists.

Let $L_P \subseteq L$ be the set of locks which cannot be opened by the pair of scientists in P. Note that for all $P \in T$, $L_P \neq \varnothing$.

Consider $P, Q \in T$ with $P \neq Q$, then $|P \cup Q| \geq 3$. Thus, $L_P \cap L_Q = \varnothing$. Now, define the function

$$f : T \to L$$

such that $f(P) = \ell_P$ where ℓ_P is an arbitrary element from L_P (note that we can do this without using Choice as each of the L_P's is finite).

Suppose that there exist $P, Q \in T$ with $P \neq Q$ such that $f(P) = f(Q) = \ell_0$. Notice that this would mean that $\ell_0 \in L_P \cap L_Q$, which is a contradiction. Therefore, f must be injective. Hence,

$$|L| \geq |T| = C_2^6 = 15.$$

Now, choose an arbitrary element from S, say s_k, and consider the set $S' = S - \{s_k\}$. Let T' be the set of all subsets of S' with 2 elements, then for any $P \in T', L_{P \cup \{s_k\}} = \varnothing$, i.e., s_k must have the keys to open all locks which the pair P cannot open with their own keys. Let this set of keys be K_P. Note that $L_P \cap L_Q = \varnothing$ whenever $P \neq Q$, so $K_P \cap K_Q = \varnothing$ whenever $P \neq Q$. Let the set of all keys held by s_k be H_k, then

$$|H_k| \geq |S'| = C_2^5 = 10.$$

1.2   Permutations

A fundamental problem in combinatorics is described as follows: given a set S, how many ways are there to arrange r elements in S, i.e. how many **distinct sequences** can be formed using the elements in S without repetition? The process of selecting elements from S and arranging them as a sequence is known as **permutation**.

Note that forming a sequence using r elements from a set S is an event consisting of r stages, as we need to select an element for each of the r terms of the sequence. Suppose S has n elements. For the first term of the sequence, we can choose any of the elements in S, so there is n ways to do it. For the second term, since we cannot repeat the elements, we are left with $n - 1$ choices.

Continue choosing elements in this way, we realise that if we choose the terms sequentially, when we reach the k-th term we will be left with $n - k + 1$ options as the previous $(k - 1)$ terms have taken away $(k - 1)$ elements. By Theorem 1.1.2, we know that the number of sequences which can be formed is given by $\prod_{i=1}^{r}(n - r + i)$.

Definition 1.2.1 ▶ Permutations

Let A be a finite set such that $|A| = n$, an r-permutation of A is a way to arrange r

elements of A, denoted as P_r^n and given by

$$P_r^n = \prod_{i=1}^{r}(n - r + i) = \frac{n!}{(n-r)!}.$$

With some algebraic manipulations, it is easy to derive the following formula, which we, however, will prove in a combinatorial manner.

Proposition 1.2.2 ▸ An Identity for Permutations

Let $n, r \in \mathbb{N}$ with $r \leq n$, then $P_r^{n+1} = P_r^n + rP_{r-1}^n$.

Proof. Let $S = \{x \in \mathbb{N}^+ : x \leq n + 1\}$ represent $(n + 1)$ distinct objects. Consider a permutation of S:

If $n + 1$ is not inside the permutation, this is equivalent to an r-permutation of $S - \{n + 1\}$, so there are P_r^n such permutations.

If $n + 1$ is inside the permutation, it means we need to first find an $(r - 1)$-permuation of $S - \{n + 1\}$, which has P_{r-1}^n ways to do. After that, we need to insert $n + 1$ into each of these $(r - 1)$-permutations. Note that for each of such permutations, there are r positions into which we can place $n + 1$. Therefore, the total number of r-permuations of S derived in this manner is rP_{r-1}^n.

Therefore, there are $P_r^n + rP_{r-1}^n$ r-permuations of S, i.e. $P_r^{n+1} = P_r^n + rP_{r-1}^n$. $\qquad\square$

Consider the following scenario: suppose $T_1, T_2, \cdots T_n$ are distinct labels and $a_1, a_2, \cdots, a_r$ are r distinct objects. If we wish to associate each object with one label, then each of the ways to do so is a permutation, the terms of which are taken from the T_i's. Notice that under this setting, the T_i's are allowed to be repeated in a permutation. To reason such problems, we first introduce the notion of a *multi-set*.

Definition 1.2.3 ▸ Multi-set

The **multi-set** is defined to be

$$M := \{r_1 \cdot a_1, r_2 \cdot a_2, \cdots, r_n \cdot a_n\},$$

where a_i has r_i identical copies.

Observe that in a permutation of such a multi-set M, the object a_i is repeated r_i times. This brings us to the following formula:

Proposition 1.2.4 ▶ Generalised Formula for Permutations

Let $k \in \mathbb{N}^+$ and let $A_1, A_2, \cdots, A_k$ be k distinct objects, where A_i occurs $n_i > 0$ times for $i = 1, 2, \cdots, k$, then each permutation for these objects, corresponds to a unique permutation of the multi-set

$$M = \{n_1 \cdot A_1, n_2 \cdot A_2, \cdots, n_k \cdot A_k\},$$

and the total number of permutations is given by

$$\frac{\left(\sum_{i=1}^{k} n_i\right)!}{\prod_{i=1}^{k} (n_i!)}.$$

We may also consider the special case where we want to form an r-permutation with k objects, where each object can repeat an arbitrary number of times. Notice that this is equivalent to finding the permutations of the multi-set

$$M = \{\infty \cdot a_1, \infty \cdot a_2, \cdots, \infty \cdot a_k\}.$$

In this case, we can simply iterate through each of the slots in the r-permutation and choose the object to be placed there. Trivially, there are always k choices for each slot.

Proposition 1.2.5 ▶ A Special Case

Consider the multi-set

$$M = \{\infty \cdot a_1, \infty \cdot a_2, \cdots, \infty \cdot a_n\}.$$

The number of r-permutations formed using the elements from M is given by n^r.

Consider arranging n distinct objects around a circle. If the slots around the circle are uniquely labelled, this is exactly the same as permutations along a straight line.

However, if the slots are identical, i.e. we are arranging n distinct objects around a circle with identical slots, only the **relative positions** of the objects matter.

Let $\boldsymbol{x}_i$ be an arbitrary straight-line permutations of the n objects and let $\boldsymbol{y}_i$ be the corresponding circular permutation of the n objects.

Note that if we translate every element in $\boldsymbol{x}_i$ by k positions, this will result in a different straight-line permutation $\boldsymbol{x}_j$ but does not change the corresponding circular permutation because the relative positions of the objects remain unchanged.

Notice that k can take the values $0, 1, 2, \cdots, n-1$, so for the same set of n distinct objects,

every circular permutation is mapped to n straight-line permutations.

Definition 1.2.6 ▸ Circular Permutations

Let A be a finite set such that $|A| = n$, a circular r permutation of A is a way to arrange r elements of A around a circular locus, denoted as Q_r^n and given by

$$Q_r^n = \frac{P_r^n}{r} = \frac{n!}{r(n-r)!}.$$

1.3   Combinations

Beside permutations, there are also occasions where we only care about which elements from a particular set are selected instead of the order of selection. Note that if we want to find a selection of r elements from a set A where the order of selected elements does not matter, it is equivalent to finding a subset of A containing r elements. This motivates us to give the following definition:

Definition 1.3.1 ▸ Combinations

Let A be a finite set such that $|A| = n$, an r-combination of A is a set $B \subseteq A$ with $|B| = r$. The number of combinations of A is given by

$$C_r^n = \frac{P_r^n}{P_r^r} = \frac{n!}{r!(n-r)!} = \binom{n}{r}.$$

Remark. Two obvious results:
1. If $r > n$ or $r < 0$, $C_r^n = 0$;
2. $C_r^n = C_{n-r}^n$.

Similar to permutations, we have the following important identity:

Theorem 1.3.2 ▸ Pascal's Triangle

Let n be an integer with $n \geq 2$ and let r be an integer with $0 \leq r \leq n$, then

$$C_r^{n+1} = C_{r-1}^n + C_r^n.$$

Proof. Let $S = \{x \in \mathbb{N}^+ : x \leq n+1\}$ be $(n+1)$ distinct objects. Consider any r-combination T of S. If $n+1 \notin T$, this is equivalent to an r-combination of $S-\{n+1\}$, so there are C_r^n such permutations. If $n+1 \in T$, it suffices to find an $(r-1)$-combination

---

of $S-\{n+1\}$, which has C_{r-1}^n ways to do. Therefore, there are $C_r^n + C_{r-1}^n$ r-combinations of S, i.e. $C_r^{n+1} = C_r^n + C_{r-1}^n$. □

A useful application of combinations, derived directly from the definition, is to count the number of subsets for a given set which is finite. In other words, if we are given a set A with $|A| = n \in \mathbb{N}$, we wish to find a general formula for $|\mathcal{P}(A)|$. Let A_i be the set of all subsets of A whose cardinality is i, then clearly

$$|\mathcal{P}(A)| = \sum_{i=0}^n |A_i| = \sum_{i=0}^n C_i^n.$$

We can expand the above expression algebraically and realise that it simplifies to 2^n. However, in a combinatorial perspective, we can prove this result in a more succinct manner:

Theorem 1.3.3 ▶ General Formula for $|\mathcal{P}(A)|$

Let A be a finite set. If $|A| = n$, then $|\mathcal{P}(A)| = 2^n$.

Proof. Let S be an arbitrary subset of A. Consider an arbitrary element $a \in A$, then either $a \in S$ or $a \notin S$. Let $a_i \in A$ for $i = 1, 2, \cdots, n$. For all $S \in \mathcal{P}(A)$, We replace a_i by 1 if $a_i \in S$, and by 0 otherwise. Let B be the set of all binary sequences of length n. It is clear that there exists a bijection between $\mathcal{P}(A)$ and B, and so $|\mathcal{P}(A)| = |B|$. For each binary sequence of length n, each of its digits is either 0 or 1. By Theorem 1.1.2, this means that there are in total 2^n such binary sequences. Therefore,

$$|\mathcal{P}(A)| = |B| = 2^n.$$

□

Given a set of n distinct objects, we wish to find the number of combinations taken from the set where any element is allowed to be selected for multiple times. To reason about such problems, we introduce the following notion of a *multi-subset*.

Definition 1.3.4 ▶ Multi-subset

Let $M = \{\infty \cdot a_1, \infty \cdot a_2, \cdots, \infty \cdot a_n\}$ be a multi-set where $n \in \mathbb{N}$. An r-element **multi-subset** of M is the set

$$\{m_1 \cdot a_1, m_2 \cdot a_2, \cdots, m_n \cdot a_n\}$$

where each of the m_i's is a non-negative integer such that $\sum_{i=1}^n m_i = r$. We denote the number of r-element multi-subsets of M by H_r^n.

Intuitively, we can view each of the a_i's as a "box" which holds m_i balls. Therefore, if we

can find the ways to distribute a total of r balls into the n "boxes", each of the distribution will then correspond to a multi-subset of M!

Proposition 1.3.5 ▸ A Formula for H_r^n

Let $M = \{\infty \cdot a_1, \infty \cdot a_2, \cdots, \infty \cdot a_n\}$, then the number of r-element multi-subsets of M is given by

$$H_r^n = C_r^{r+n-1}.$$

Proof. Consider a binary string formed using r zeros. Note that we can insert a number of ones into the binary string to partition the zeros into different sections. Now, suppose we insert $(n-1)$ ones into the binary string, it will result in a binary string containing r zeros and $(n-1)$ ones, such that the zeros are partitioned into n sections.

We can then number the sections using $1, 2, 3, \cdots, n$. For the i-th section, the number of zeros is recorded as m_i. Notice that in this case, each of the binary strings will correspond to a multi-subset in the form of

$$\{m_1 \cdot a_1, m_2 \cdot a_2, \cdots, m_n \cdot a_n\}.$$

One can check that this will establish a bijection between the set of all multi-subsets containing n distinct types of objects and the set of all binary strings with r zeros and $(n-1)$ ones. Therefore, the above has proven that $H_r^n = C_r^{r+n-1}$. □

1.4 Binomial and Multinomial Coefficients

In previous sections on combinations, we have already introduced the ways to choose r items from a collection of n distinct items, the number of which is given by

$$C_r^n = \binom{n}{r}.$$

In this section, we will see that there is a clear relation between the number of r-combinations to the *binomial expansion*.

Theorem 1.4.1 ▸ Binomial Expansion

Let $n \in \mathbb{N}$, then

$$(x+y)^n = \sum_{i=0}^{n} \left[\binom{n}{i} x^{n-i} y^i \right].$$

Proof. Note that

$$(x + y)^n = \prod_{i=1}^{n} (x + y).$$

Let the coefficient of $x^{n-r} y^r$ be k, where r is an integer with $0 \leq r \leq n$. Notice that k is exactly the number of ways to choose $(n - r)$ copies of x's from the distinct n terms of $(x + y)$, which is given by

$$\binom{n}{n-r} = \binom{n}{r}.$$

Therefore, summing up the terms, we have

$$(x + y)^n = \sum_{i=0}^{n} \left[\binom{n}{i} x^{n-i} y^i \right].$$

$\square$

There are many nice algebraic properties about the binomial coefficient $\binom{n}{r}$. Beside Theorem 1.3.2, the following can be proven with a few simple algebraic manipulations:

1. $\binom{n}{r} = \frac{n}{r} \binom{n-1}{r-1}.$

2. $\binom{n}{r} = \frac{n-r+1}{r} \binom{n}{r-1}.$

3. $\binom{n}{m} \binom{m}{r} = \binom{n}{r} \binom{n-r}{m-r}.$

One important identity for binomial coefficients is as follows:

Theorem 1.4.2 ▶ Vandermonde's Identity

Let $m, n, r \in \mathbb{N}$, then

$$\sum_{i=0}^{r} \left[\binom{m}{i} \binom{n}{r-i} \right] = \binom{m+n}{r}.$$

Proof. Consider the set

$$X = \{a_1, a_2, \cdots, a_m, b_1, b_2, \cdots, b_n\}.$$

Clearly, $|X| = m+n$. Let $A \subseteq X$ such that $|A| = r$, we consider the number of such A's.

Let i be an integer with $0 \leq i \leq r$. If A contains exactly i elements from the a_j's, then it will contain exactly $(r-1)$ elements from the b_j's. Therefore, the number of A's for each $i = 1, 2, \cdots, r$ is given by

$$\binom{m}{i}\binom{n}{r-i}.$$

However, we know that the total number of such A's is just C_r^{m+n}, so we have

$$\sum_{i=0}^{r} \left[\binom{m}{i}\binom{n}{r-i} \right] = \binom{m+n}{r}.$$

$\square$

Recall that in Theorem 1.3.2, we essentially established a recurrence relation for the binomial coefficients. Thus in theory, it is possible to generate any binomial coefficient recursively. The following theorem illustrates this.

Theorem 1.4.3 ▶ Chu Shih-Chieh (CSC) Identity

Let $r, n, k \in \mathbb{N}$ with $n \geq r$, then

$$\sum_{i=0}^{n-r} \binom{r+i}{r} = \binom{n+1}{r+1}, \qquad \sum_{i=0}^{k} \binom{r+i}{i} = \binom{r+k+1}{k}.$$

Proof. Consider the set

$$X = \{x \in \mathbb{N}^+ : x \leq n+1\}.$$

We will count the number of $(r+1)$-element subsets of X in the following way:

Fix k to be the smallest element in some $Y \subseteq X$, then it suffices to choose r elements from $X - \{x \in \mathbb{N}^+ : x \leq k\}$. Note that the maximum for k is $n - r + 1$, and so there are $n - r + 1$ disjoint cases. Therefore, it is easy to see that the total number of subsets is given by

$$\sum_{i=0}^{n-r} \binom{r+i}{r}.$$

However, we know that this is equivalent to choosing $(r+1)$ elements directly from

X, so

$$\sum_{i=0}^{n-r} \binom{r+i}{r} = \binom{n+1}{r+1}.$$

Note that $C_r^{r+i} = C_i^{r+i}$ and $C_{r+1}^{n+1} = C_{n-r}^{n+1}$. Therefore, by setting $k = n - r$, it is obvious by symmetry that

$$\sum_{i=0}^{k} \binom{r+i}{i} = \binom{r+k+1}{k}.$$

$\square$

Binomial coefficients can be generalised as *multinomial coefficients*, which has a nice correspondence to a special variant of distribution problems.

> **Definition 1.4.4 ▶ Multinomial Coefficient**
>
> Let $m, n \in \mathbb{N}$, with $m \geq 1$. The **multinomial coefficient**
>
> $$\binom{n}{n_1, n_2, \cdots, n_m} = \frac{n!}{\prod_{i=1}^{m} n_i!}$$
>
> is the number of ways to distribute n distinct objects into m distinct collections such that there are exactly n_i objects in the i-th collection.

Observe that the binomial coefficients are just a special case for multinomial coefficients with $m = 2$ and $n_1 + n_2 = n$. Respectively, multinomial coefficients also generalise binomial expansion.

> **Theorem 1.4.5 ▶ Multinomial Expansion**
>
> *For $m, n \in \mathbb{N}+$, we have*
>
> $$\left(\sum_{i=1}^{m} x_i \right)^n = \sum_{\sum_{j=1}^{m} n_j = n} \binom{n}{n_1, n_2, \cdots, n_m} \prod_{i=1}^{m} x_i^{n_i}.$$

1.5   Principle of Inclusion and Exclusion (PIE)

Consider the following naïve question:

> If $|A| = a$ and $|B| = b$, what is $|A \cup B|$?

The solution is simple: we first count how many elements are in A and then count how

many elements are in B, but note that there are some elements which might be in both sets and are thus counted twice, so we need to subtract these elements away.

Proposition 1.5.1 ▶ Cardinality of the Union of Two Finite Sets

Let A and B be two finite sets, then

$$|A \cup B| = |A| + |B| - |A \cap B|.$$

In particular, if A and B are disjoint, then $|A \cup B| = |A| + |B|$.

Proposition 1.5.1 can be generalised for a finite sequence of finite sets.

Theorem 1.5.2 ▶ Principle of Inclusion and Exclusion

Let $E_1, E_2, \cdots, E_n$ be a sequence of finite sets. In general, we have

$$\left| \bigcup_{i=1}^{n} E_i \right| = \sum_{j=1}^{n} \left[(-1)^{j+1} \left(\sum_{1 \le k_1 \le k_2 \le \cdots \le k_j \le n} \left| \bigcap_{r=1}^{j} E_{k_r} \right| \right) \right].$$

Proof. Define a function $f_S : S \to \{0, 1\}$ by

$$f_S(x) = \begin{cases} 1 & \text{if } x \in S \\ 0 & \text{if } x \notin S \end{cases}.$$

Let $E = \bigcup_{i=1}^{n} E_i$. Consider

$$\prod_{i=1}^{n} \left(f_E(x) - f_{E_i}(x) \right). \tag{*}$$

For any x, if $x \in E$, then $x \in E_k$ for some $k \in \{x \in \mathbb{N} : x \le n\}$, which means that $f_E(x) - f_{E_k}(x) = 0$; if $x \notin E$, then $f_E(x) = f_{E_i}(x) = 0$ for all $i \in \{x \in \mathbb{N} : x \le n\}$. In either case, (*) is identically 0.

Note that for any sets A, B, $f_{A \cap B}(x) = f_A(x) f_B(x)$. In particular, $f_{E \cap E_i}(x) = f_{E_i}(x)$ for all $i = 1, 2, \cdots, n$. Expanding (*), we have

$$f_E(x) = \left[f_E(x) \right]^n = \sum_{j=1}^{n} \left\{ (-1)^{j+1} \left[\sum_{1 \le k_1 \le k_2 \le \cdots \le k_j \le n} \left(\prod_{r=1}^{j} f_{E_{k_r}}(x) \right) \right] \right\}.$$

Summing up $f_E(x)$ over all x leads to the identity in the original statement. □

We can generalise this further. Note that every set is uniquely determined by some formula ϕ with some parameter p. This means that the statement $x \in X$ is logically equiva-

lent to "x satisfies a property $\phi(x, p)$". This leads to the general principle of inclusion and exclusion.

Theorem 1.5.3 ▸ General Principle of Inclusion and Exclusion

Let S be a finite set with $|S| = n$ and let $\{P_1, P_2, \cdots P_q\}$ be a set of properties. Suppose that $E(m)$ is the number of elements in S that satisfy exactly m of the q properties and that $\omega(P_{i_1} P_{i_2} \cdots P_{i_m})$ is the number of elements in S that satisfy exactly $P_{i_1}, P_{i_2}, \cdots, P_{i_m}$, such that

$$\omega(m) = \sum \omega(P_{i_1} P_{i_2} \cdots P_{i_m}),$$

then

$$E(m) = \sum_{k=m}^{q} \left[(-1)^{k-m} \binom{k}{m} \omega(k) \right].$$

Proof. Let $x \in S$ be an arbitrary element which satisfies exactly t properties. If $t < m$ or $q < t$, then x contributes a count of 0 to both sides. If $t = m$, then x contributes a count of 1 to $E(m)$ and to $\omega(m)$ and contributes a count of 0 to $\omega(k)$ for all $k > m$.

If $q \geq t > m$, then x contributes a count of 0 to the left-hand side. On the right-hand side, x is counted $\binom{t}{k}$ times for $k = m, m+1, \cdots, t$. So the total contribution of x to the right-hand side is

$$\sum_{k=m}^{t} \left[(-1)^{k-m} \binom{t}{k} \binom{k}{m} \right] = \sum_{k=m}^{t} \left[(-1)^{k-m} \binom{t}{m} \binom{t-m}{k-m} \right]$$

$$= \sum_{k=0}^{t-m} \left[(-1)^{k} \binom{t}{m} \binom{t-m}{k} \right]$$

$$= \binom{t}{m} \sum_{k=0}^{t-m} \left[(-1)^{k} \binom{t-m}{k} \right]$$

$$= 0.$$

Therefore, every $x \in S$ contributes an equal count to both sides of the identity, so the identity holds. □

In particular, we see that

$$E(0) = \sum_{i=0}^{q} \left[(-1)^{i} \omega(i) \right].$$

1.6   Distribution Problems

Another problem in which we are interested is to **distribute** the objects from a collection into a finite number of sub-collections. Specifically, we consider the number of ways to

1. distribute r distinct objects into n identical collections;

2. distribute r distinct objects into n distinct collections;

3. distribute r identical objects into n identical collections;

4. distribute r identical objects into n distinct collections.

These 4 basic models of distribution problems give rise to many variations. We will only discuss the first two types of distribution problems here. The easiest case for this distribution problem is that each collection contains at most one object, i.e., each collection either contains an object, or contains no object at all.

Proposition 1.6.1 ▶ Distinct Objects into Singleton Distinct Collections

The number of ways to distribute r distinct objects into n distinct collections such that each collection contains at most 1 object is given by P_r^n.

We then consider the case where each collection can contain any number of objects.

Proposition 1.6.2 ▶ Distinct Objects into Distinct Collections

The number of ways to distribute r distinct objects into n distinct collections such that each collection can contain any number of objects is given by

$$\frac{(n + r - 1)!}{(n - 1)!}.$$

Proof. Left to the reader as an exercise.   □

For distributing distinct objects into identical collections, the reasoning becomes a bit more complicated. We first consider a scenario where the collections are circular.

Definition 1.6.3 ▶ Stirling Numbers of the First Kind

Let $r, n \in \mathbb{N}$ such that $0 \leq n \leq r$, then the **Stirling Number of the First Kind**, $s(r, n)$, is the number of ways to arrange r **distinct** onjects around n **identical** circles such that no circle is empty.

Remark. Some obvious results:
1. $s(r, 0) = 0$ if $r \geq 1$.
2. $s(r, r) = 1$ if $r \geq 0$.
3. $s(r, 1) = (r - 1)!$ if $r \geq 2$.
4. $s(r, r - 1) = C_2^r$ if $r \geq 2$.

One thing to take note of is that there is no general algebraic formula for $s(r, n)$. Instead, all Stirling Numbers of the First Kind follow a recurrence relation in 2 parameters.

Theorem 1.6.4 ▶ A Recurrence Relation for $s(r, n)$

For $r, n \in \mathbb{N}^+$ and $r \leq n$, we have

$$s(r, n) = s(r - 1, n - 1) + (r - 1)s(r - 1, n).$$

Proof. Consider the set $\{x_1, x_2, \cdots, x_r\}$ to be the r distinct objects. We consider two cases. If x_r is distributed to a circle such that it is the only objects around that circle, it suffices to find the number of ways to arrange the rest $(r - 1)$ distinct objects around the rest $(n - 1)$ identical circles, which there are $s(r - 1, n - 1)$ ways to do.

If x_r is adjacent to some other object, we can first arrange the rest $(r - 1)$ distinct objects around the n identical circles, which there are $s(r - 1, n)$ ways to do. After that, we can choose any one of the $(r - 1)$ spaces between two adjacent objects to slot in x_r. Therefore, there are $(r - 1)s(r - 1, n)$ ways to distribute the objects. By Theorem 1.1.1, the total number of distributions is

$$s(r, n) = s(r - 1, n - 1) + (r - 1)s(r - 1, n).$$

□

Next, we consider the case where the collections arrange their elements linearly.

Definition 1.6.5 ▶ Stirling Number of the Second Kind

The **Stirling Number of the Second Kind**, denoted as $S(r, n)$, is the number of ways to distribute r distinct objects into n indentical collections such that no collection is empty.

> *Remark.* Some trivial results:
> - $S(0,0) = 1$.
> - $S(r,0) = S(0,n) = 0$ for all $r, n \in \mathbb{N}^+$.
> - $S(r,n) > 0$ for all $r, n \in \mathbb{N}$ with $r \geq n \geq 1$.
> - $S(r,n) = 0$ if $n > r \geq 1$.
> - $S(r,1) = S(r,r) = 1$ for all $r \in \mathbb{N}^+$.

Based on the above trivial results, we can use the following recurrence relation to derive a general formula for $S(r,n)$ for any r and n.

Theorem 1.6.6 ▶ Recurrence Relation of $S(r,n)$

For all $r, n \in \mathbb{N}$ with $r \geq n$,

$$S(r,n) = S(r-1, n-1) + nS(r-1, n).$$

Proof. Consider the first object. If the first object is alone in a collection, it is equivalent to distributing $(r-1)$ distinct objects into $(n-1)$ identical collections such that no collection is empty, which can be done in $S(r-1, n-1)$ ways. If the first object is not alone, we first distribute the rest $(r-1)$ distinct objects into the n identical collections in $S(r-1, n)$ ways, and then choose a collection to place the first object into in n ways, which can be done in $nS(r-1, n)$ ways. Therefore,

$$S(r,n) = S(r-1, n-1) + nS(r-1, n).$$

□

The Stirling numbers of the second kind are closely related to the problem of set partition.

Definition 1.6.7 ▶ Partition

Let A be a set, an n-partition of A is a set $S \subset \mathcal{P}(A)$ such that for all $S_1, S_2 \in S$ with $S_1 \neq S_2$, we have $S_1 \cap S_2 = \varnothing$ and $\bigcup S = A$.

Note that in a set, the elements are pairwise distinct. Therefore, a partition of A into i disjoint subsets is equivalent to a distribution of $|A|$ distinct objects into i identical collections. Therefore, the number of different partitions of A is just

$$\sum_{i=1}^{|A|} S(|A|, i).$$

To find an exact formula for $S(r,n)$, we consider the following proposition:

> **Proposition 1.6.8 ▶ Number of Surjective Mappings**
>
> *Let $F(r, n)$ where $r, n \in \mathbb{N}^+$ deonote the number of surjective mappings*
>
> $$f : [r] - \{0\} \rightarrow [n] - \{0\},$$
>
> *then*
>
> $$F(r, n) = \sum_{k=0}^{n} \left[(-1)^k \binom{n}{k} (n-k)^r \right].$$

Proof. Note that f is a mapping from a domain of r elements to a co-domain of n elements. Let M_k be the set of such mappings by which at least k elements in the co-domain do not have pre-images. Note that for each element in the domain, its image can be any of the remaining $(n - k)$ elements. Therefore,

$$|M_k| = \binom{n}{k} (n-k)^r.$$

By Theorem 1.5.3,

$$F(r, n) = \left| \left(\bigcup_{k=0}^{n} M_k \right)' \right| = \sum_{k=0}^{n} \left[(-1)^k \binom{n}{k} (n-k)^r \right].$$

□

Note that $F(r, n)$ is just the number of ways to distribute r distinct elements into n collections where the order of the collections matters, so $F(r, n) = n! S(r, n)$. This gives rise to some corollaries, for example:

$$S(r, n) = \frac{1}{n!} \sum_{k=0}^{n} (-1)^k \binom{n}{k} (n-k)^r$$

Since $S(r, n) = 0$ if $n > r \geq 1$, we have

$$\sum_{k=0}^{n} (-1)^k \binom{n}{k} (n-k)^r = 0 \quad \text{if } n > r \geq 1.$$

A common application of distribution problems with n identical objects and m identical collections is the partition of an integer.

> ### Definition 1.6.9 ▶ Integer Partition
>
> Let $n \in \mathbb{N}^+$. A **partition** of n is a non-increasing sequence of positive integers whose sum is n.

A useful tool to study such partitions is *Ferrers diagram*.

> ### Definition 1.6.10 ▶ Ferrers Diagram
>
> Let $n \in \mathbb{N}^+$ and consider a partition $P = \{n_1, n_2, \cdots, n_k\}$. The **Ferrers diagram** of the partition, denoted as $\mathcal{F}(P)$, is an array of left-justified asterisks in which the i-th row has the length of n_i.

Just like matrices, a Ferrers diagram can be transposed.

> ### Definition 1.6.11 ▶ Conjugate Partition
>
> Let $\mathcal{F}(P)$ be the Ferrers diagram of an integer partition P. Q is known as the **conjugate partition** of P if $\mathcal{F}(Q) = \mathcal{F}(P)^{\mathrm{T}}$.

Intuitively, conjugate partition is a **symmetric relation**, and if P and Q are conjugate partitions, the number of parts of Q is just the size of the largest part of P. This leads to the following theorem:

> ### Proposition 1.6.12 ▶ Size of Conjugate Partitions
>
> *Let $k \leq r \in \mathbb{N}^+$. The number of partitions of r into k parts equals the number of partitions of r with the largest part having size k.*
>
> ---
>
> *Proof.* Let $\mathcal{P}$ be the family of partitions of r into k parts and $\mathcal{Q}$ be the family of partitions of r whose largest part has size k. Define a function $f : \mathcal{P} \rightarrow \mathcal{Q}$ such that $\mathcal{F}(f(X)) = \mathcal{F}(X)^{\mathrm{T}}$, i.e., $f(X)$ and X are conjugates. It is easy to see that f is well-defined and bijective, so $|\mathcal{P}| = |\mathcal{Q}|$. $\qquad\square$

Proposition 1.6.12 leads to the following corollary:

> ### Corollary 1.6.13 ▶ Integer Partition into Limited Parts
>
> *Let $k \leq r \in \mathbb{N}^+$. The number of partitions of r into at most k parts equals the number of partitions of r with the largest part having size at most k.*

The following formula can de derived:

$$\sum_{k=1}^{m} p(n,k) = p(n+m, m).$$

1.7 Pigeonhole Principle

Theorem 1.7.1 ▶ Pigeonhole Principle

Let $k, n \in \mathbb{N}^+$. If at least $kn + 1$ distinct objects are distributed into n distinct sets, then there exists a set K such that $|K| \geq k + 1$.

Proof. Let $a_1, a_2, \cdots, a_{kn+1}$ be the $kn + 1$ distinct objects to be distributed into the sets $K_1, K_2, \cdots, K_n$. Suppose there is no $i \in \mathbb{N}^+$ such that $|K_i| \geq k + 1$, i.e. $K_i \leq k$ for all $i = 1, 2, \cdots, n$. This means that $\sum_{i=1}^{n} |K_i| \leq kn < kn + 1$, which is a contradiction. $\square$

Note that Pigeonhole Principle only guarantees existence rather than uniqueness. We may also consider the following generalised version:

Theorem 1.7.2 ▶ Generalised Pigeonhole Principle

Let $n, k_1, k_2, \cdots, k_n \in \mathbb{N}^+$. If at least $\left[\sum_{i=1}^{n} k_i + (n-1)\right]$ distinct objects are distributed into n distinct sets $S_1, S_2, \cdots, S_n$, then there exists at least one S_i for $i = 1, 2, \cdots, n$ such that $|S_i| \geq k_i$.

Proof. Suppose there is no $i \in \mathbb{N}^+$ such that $|S_i| \geq k_i$, i.e., $|S_i| \leq k_i - 1$ for any $i = 1, 2, \cdots, n$. This means that $\sum_{i=1}^{n} |K_i| \leq \sum_{i=1}^{n} (k_i - 1) = \sum_{i=1}^{n} k_i - n$, which is a contradiction. $\square$

Note that by setting $k_1 = k_2 = \cdots = k_n = 2$ we get from Theorem 1.7.2 to Theorem 1.7.1.

The Pigeonhole Principle is closely linked to the *Ramsey Types of Problems*, which are motivated by the following question statement:

Example 1.7.3 ▶ Monochromatic Triangles with Six Points

Let $A_1, A_2, A_3, A_4, A_5, A_6$ be six arbitrary points in a plane. We join these points pairwise with either a **red** line segment or a **blue** line segment. Prove that after each point has been joined with any other point, there exists at least one **blue** triangle or at least one **red** triangle whose vertices are three of the six points.


Proof. Consider the line segments joining A_h with the other five points. Note that this is equivalent to colouring 5 line segments using either red or blue. By Theorem 1.7.1, this means that at least 3 of these line segments have the same colour. Without loss of generality, suppose the line segments $A_h A_i$, $A_h A_j$ and $A_h A_k$ are red. Consider the line segments $A_i A_j$, $A_j A_k$ and $A_i A_k$. If they are all blue, then we will have a blue triangle. Otherwise, if any one of them is red, then those two vertices together with A_h will form a red triangle. $\square$

We can extend the above question a bit. Suppose we have 17 arbitrary points $X_1, X_2, \cdots, X_{17}$ in a plane and we join these points in the same way as Example 1.7.3 but with three distinct colours: **red**, **blue** and **orange**. Can we still guarantee the existence of at least on triangle whose sides are all of the same colour?

It turns out that we can! Again we can consider all line segments joining X_1 with the other 16 vertices. By Theorem 1.7.1, we must have at least 6 of them having the same colour. Without loss of generality, assume that these 6 line segments are all red.

Now, examine these 6 vertices joined to X_1 with the same-coloured line segments. If any line segment between these 6 vertices is red, then we have a red triangle. Otherwise, the line segments joining these 6 vertices are coloured with only two colours, and by Example 1.7.3 we are guaranteed to have a triangle with all three sides having the same colour as well.

Going back to Example 1.7.3, one might notice that 6 is the smallest possible number of vertices for us to guarantee the existence of such a triangle. In the case of 5 vertices, consider:
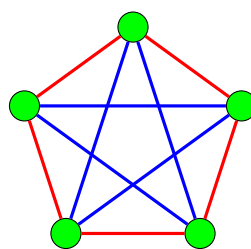
This motivates the following definitions:

> ### Definition 1.7.4 ▸ Clique
>
> A **clique** is defined to be a finite set of vertices such that every pair of vertices are adjacent to each other (i.e., there exists a line segment joining any two vertices). A clique with k vertices is called a k-clique.

> **Definition 1.7.5 ▶ Ramsey Number**
>
> Given $p, q \in \mathbb{N}^+$. The **Ramsey Number** $R(p, q)$ is the smallest natural number n such that for any colouring of the edges of an n-clique by 2 colours, there exists either a monochromatic p-clique or a monochromatic q-clique.

> *Remark.* Some obvious results:
> - $R(p, q) = R(q, p)$.
> - $R(1, q) = 1$.
> - $R(2, q) = q$.

These develop into what we now know as *Ramsey Theory* which probes into the question:

> How many elements of a particular structure S must there exist to guarantee that a certain property p will always hold for S?

It turns out that calculating the Ramsey Numbers can be extremely difficult. However, we do have a way to establish an upper bound for each of the Ramsey Numbers.

> **Theorem 1.7.6 ▶ Recurrence Relation of Ramsey Numbers**
>
> *For all integers $p, q \geq 2$,*
>
> $$R(p, q) \leq R(p - 1, q) + R(p, q - 1).$$
>
> *In particular, if both $R(p - 1, q)$ and $R(p, q - 1)$ are even, then*
>
> $$R(p, q) \leq R(p - 1, q) + R(p, q - 1) - 1.$$
>
> *Proof.* Let $n = R(p - 1, q) + R(p, q - 1)$. To prove that $R(p, q) \leq n$, it suffices to prove that for any n-clique, any colouring of its edges using blue and red will result in either a blue p-clique or a red q clique. Consider an arbitrary vertex V of any n-clique. Note that V is incident to $(R(p - 1, q) + R(p, q - 1) - 1)$ vertices and edges. By Theorem 1.7.2 this means that either at least $R(p - 1, q)$ edges are blue or at least $R(p, q - 1)$ edges are red.
>
> Without loss of generality, consider the $R(p - 1, q)$-clique whose vertices are joined to V by blue edges. By Definition 1.7.5, this clique contains either a blue $(p - 1)$-clique or a red p-clique. If it contains a red p-clique, then we are done. If it contains a blue $(p - 1)$-clique, this clique together with V will form a blue p-clique.

Either way, it is guaranteed that

$$R(p, q) \leq n = R(p - 1, q) + R(p, q - 1).$$

Now we proceed to proving the case where $R(p - 1, q)$ and $R(p, q - 1)$ are both even. Let $m = R(p - 1, q) + R(p, q - 1) - 1$. Fix an arbitrary vertex W, then it suffices to prove that among the $(m - 1)$ edges incident to W, either at least $R(p - 1, q)$ are blue or at least $R(p, q - 1)$ are red.

Suppose on contrary that there are exactly $(R(p - 1, q) - 1)$ blues edges and exactly $(R(p, q - 1) - 1)$ red edges for all vertices in the m-clique. This means that the total number of blue edges in the m-clique is

$$N = \frac{m\,(R(p - 1, q) - 1)}{2}.$$

However, m and $R(p - 1, q) - 1$ are both odd, this means that N is not an integer, which is impossible. □

The above recurrence allows us to prove the following theorem:

Theorem 1.7.7 ▶ Ramsey's Theorem

For all $p, q \in \mathbb{N}^+$, the Ramsey Number $R(p, q)$ always exists.

2

Tools for Combinatorial Problems

2.1 Ordinary Generating Functions

> **Definition 2.1.1 ▶ Ordinary Generating Function (OGF)**
>
> Let (a_r) be a sequence in $\mathbb{R}$. The **ordinary generating function** of (a_r) is defined to be the series
> $$A(x) = \sum_{i=0}^{\infty} a_i x^i.$$

Note that the set of generating functions is a vector space. In particular, if the generating functions satisfy $C(x) = A(x) + B(x)$, then $c_r = a_r + b_r$ and if $C(x) = A(x)B(x)$, then we have $c_r = \sum_{i=0}^{r} a_i b_{r-i}$.

> **Definition 2.1.2 ▶ Generalised Binomial Coefficients**
>
> Let $\alpha \in \mathbb{R}$ and $r \in \mathbb{N}$, then
> $$\binom{\alpha}{r} = \frac{\prod_{i=0}^{r-1}(\alpha - i)}{r!}.$$

Using this version of binomial coefficients, we can upgrade our binomial expansion to apply to a wider range of contexts.

> **Definition 2.1.3 ▶ Generalised Binomial Expansion**
>
> Let $\alpha \in \mathbb{R}$ then
> $$(1 \pm x)^{\alpha} = \sum_{r=0}^{\infty} \binom{\alpha}{r} (\pm x)^r.$$

We now examine a few basic ordinary generating functions. First, consider

$$a_r = \begin{cases} 1, & \text{if } r = n \\ 0, & \text{otherwise} \end{cases}.$$

It is easy to see that $A(x) = x^n$ generates this sequence.

Suppose we wish to generate the number of ways to choose r objects from n objects, i.e., consider

$$a_r = \begin{cases} \binom{n}{r}, & \text{if } 0 \le r \le n \\ 0, & \text{otherwise} \end{cases}.$$

Note that the generating function is just the binomial expansion

$$(1 + x)^n = \sum_{i=0}^{n} \binom{n}{i} x^i.$$

Now consider a special case where $a_r = 1$ forms a constant sequence. In this case we have

$$A(x) = \sum_{i=0}^{\infty} x^i = \frac{1}{1 - x}.$$

More generally, one can verify with generalised binomial expansion that the generating function for $(1, k, k^2, \cdots)$ is $\frac{1}{1-kx}$.

Notice that

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

This means that the series $(1 - x)^{-n}$ generates the number of ways to choose r objects from $(n-1+r)$ objects. In particular, if $a_r = r+1$, note that $a_r = C_r^{r+1}$, so substituting $n = 2$ yields

$$A(x) = (1 - x)^{-2} = \frac{1}{(1 - x)^2}$$

as the generating function for (a_r).

Note that an ordinary generating function is an infinite polynomial, so the set of all ordinary generating functions form a vector space.

Additionally, the following conclusions hold:

1. For any $\alpha, \beta \in \mathbb{R}$, $C(x) = \alpha A(x) + \beta B(x)$ generates $c_r = \alpha a_r + \beta b_r$.

2. $C(x) = A(x)B(x)$ generates $c_r = \sum_{i=0}^{r} a_i b_{r-i}$.

3. For $m \in \mathbb{N}^+$, $C(x) = x^m A(x)$ generates

$$c_r = \begin{cases} 0, & \text{if } 0 \le r \le m - 1 \\ a_{r-m}, & \text{if } r \ge m \end{cases}.$$

4. $C(x) = A(kx)$ generates $c_r = k^r a_r$.

5. $C(x) = (1-x)A(x)$ generates

$$
c_r = \begin{cases} a_0, & \text{if } r = 0 \\ a_r - a_{r-1} & \text{if } r \geq 1 \end{cases}.
$$

6. $C(x) = \frac{A(x)}{1-x}$ generates $c_r = \sum_{i=0}^{r} a_i$.

7. $C(x) = A'(x)$ generates $c_r = (r+1)a_{r+1}$.

8. $C(x) = xA'(x)$ generates $c_r = ra_r$.

Let us see an example of how generating functions can be used to model problems.

Technique 2.1.4 ▸ Modelling Combinations with Ordinary Generating Functions

Let $S = \{s_1, s_2, \cdots, s_n\}$ be n distinct objects and let a_r be the number of ways to select r objects from S. To model this problem using ordinary generating functions, let x^0 denote the event that s_i is not selected and $s_i x$ denote the event that s_i is selected. Consider

$$
\prod_{i=1}^{n}(1 + s_i x).
$$

Note that the number of r-combinations is essentially the number of terms in the coefficient of x^r in the expansion. Therefore, setting $s_i = 1$ for $i = 1, 2, \cdots, n$, the generating function for (a_r) is

$$
(1+x)^n = \sum_{i=0}^{n} \binom{n}{i} x^i,
$$

which is essentially the binomial expansion.

While the above modelling seems an over-kill, it becomes more useful when dealing with multi-sets. Suppose S is now a multi-set instead with s_i having c_i copies. Then, let $s_i^j x^j$ denote the event that s_i is selected j times. Following a similar process we can obtain the corresponding generating function.

In general, let

$$
M = \{n_1 \cdot b_1, n_2 \cdot b_2, \cdots, n_k \cdot b_k\}
$$

be a multi-set and let a_r be the number of ways to select r objects from M. The generating

function of (a_r) is

$$A(x) = \prod_{i=1}^{k} \left(\sum_{j=0}^{n_i} x^j \right).$$

The above method enables us to also model distribution problems of n identical objects into k distinct collections.

However, the above method will not work if the collections are also identical. Let x^i denote a collection which contains i of the identical objects and let a_r denote the number of ways to distribute n identical objects into r non-empty identical collections of size at most k, then $(x^i)^j$ denotes the event that there are j collections of size i in the distribution. As such, the generating function for (a_r) is

$$A(x) = \prod_{i=1}^{k} \left(\sum_{j=0}^{\infty} x^{ij} \right) = \frac{1}{\prod_{i=1}^{k}(1 - x^i)}.$$

This allows us to model distribution problems with various constriants.

2.2   Exponential Generating Functions

> **Definition 2.2.1 ▶ Exponential Generating Function (EGF)**
>
> Let (a_r) be a sequence in $\mathbb{R}$. The **exponential generating function** of (a_r) is defined to be the series
> $$A(x) = \sum_{i=0}^{\infty} a_i \frac{x^i}{i!}.$$

Here we introduce three basic exponential generating functions:

1. $a_r = 1$: $A(x) = \sum_{i=0}^{\infty} \frac{x^i}{i!} = e^x$.

2. $a_r = r!$: $A(x) = \sum_{i=0}^{\infty} x_i = \frac{1}{1-x}$.

3. $a_r = k^r$ for $k \neq 0$: $A(x) = \sum_{i=0}^{\infty} \frac{(kx)^i}{i!} = e^{kx}$.

Let $a_r = P_r^n$ be the number of r-permutations of n distinct objects. The exponential gener-

ating function for (a_r) is just

$$A(x) = \sum_{i=0}^{n} P_r^n \frac{x^i}{i!}$$

$$= \sum_{i=0}^{n} i! C_r^n \frac{x^i}{i!}$$

$$= \sum_{i=0}^{n} C_r^n x^i$$

$$= (1+x)^n.$$

In general, let

$$M = \{n_1 \cdot b_1, n_2 \cdot b_2, \cdots, n_k \cdot b_k\}$$

be a multi-set and let a_r be the number of r-permutations of M. The generating function of (a_r) is

$$A(x) = \prod_{i=1}^{k} \left(\sum_{j=0}^{n_i} \frac{x^j}{j!} \right).$$

2.3   Recurrence Relations

A recurrence relation, informally, is an equation that relates the n-th term of a sequence to its previous k terms.

Definition 2.3.1 ▶ Recurrence Relation

Let (a_n) be a sequence. A k**-th order recurrence relation** on (a_n) is defined to be the equation

$$a_n = f(a_{n-1}, a_{n-2}, \cdots, a_{n-k})$$

for some $k \in \mathbb{Z}$. An expression $a_n = g(n)$ which satisfies the recurrence relation for all n is known as the **solution**.

Note that to explicitly define a sequence, we need to define the first k terms of the sequence given a recurrence relation. These are known as the *initial condition*.

We will first study a most basic type of recurrence relations — *linear homogeneous* recurrecne relations.

Definition 2.3.2 ▶ **Linear Homogeneous Recurrence Relations**

A k-**th order linear homogeneous recurrence relation** of a sequence (a_n) is defined as

$$\sum_{i=0}^{k} c_i a_{n-i} = 0$$

where $c_0, c_k \neq 0$.

A *characteristic equation* of a recurrence relation of order k is obtained by replacing a_{n-i} with x^{k-i}. In the linear homogeneous case, the characteristic equation is

$$\sum_{i=0}^{k} c_i x^{k-i} = 0.$$

Let $\alpha_1, \alpha_2, \cdots, \alpha_p$ be the distinct roots to the equation with multiplicity $m_1, m_2, \cdots, m_p$ respectively, then

$$a_n = \sum_{i=1}^{p} \left[\left(\sum_{j=1}^{m_i} A_{ij} n^{j-1} \right) \alpha_i^n \right]$$

is the general solution to the recurrence relation, where the A_{ij}'s are constants. We can further generalise this to a non-homogeneous case.

Definition 2.3.3 ▶ **General Linear Recurrence Relation**

A k-**th order linear recurrence relation** of a sequence (a_n) is defined as

$$\sum_{i=0}^{k} c_i a_{n-i} = f(n)$$

where $c_0, c_k \neq 0$ and f is a function in n.

To solve a general linear recurrence relation, we first introduce the notion of a *particular solution*.

Definition 2.3.4 ▶ **Particular Solution**

Let $a_n = r(a_{n-1}, a_{n-2}, \cdots, a_{n-k})$ be a recurrence relation. A **particular solution** is defined as a function $p(n)$ that satisfies the recurrence relation.

It turns out that to solve a general linear recurrence relation, we can first convert it to the homogeneous case and find its solution known as the *complementary solution*. Augmenting the complementary solution with a particular solution will yield the general solution!

> **Theorem 2.3.5 ▸ Solution of Linear Recurrence Relations**
>
> *Let*
>
> $$\sum_{i=0}^{k} c_i a_{n-i} = f(n)$$
>
> *be a linear recurrence relation. Suppose $p(n)$ is a particular solution and $c(n)$ is the solution to the homogeneous recurrence relation*
>
> $$\sum_{i=0}^{k} c_i a_{n-i} = 0,$$
>
> *then $a_n = p(n) + c(n)$ is the general solution to the original linear recurrence relation.*
>
> *Proof.* Since $p(n)$ is a particular solution, we have
>
> $$\sum_{i=0}^{k} c_i p(n-i) = f(n).$$
>
> Furthermore, we have
>
> $$\sum_{i=0}^{k} c_i c(n-i) = 0.$$
>
> Therefore,
>
> $$\sum_{i=0}^{k} c_i \big(p(n-i) + c(n-i) \big) = f(n),$$
>
> which means $a_n = p(n) + c(n)$ is the general solution to the original linear recurrence relation. $\qquad\square$

There is no general way to derive the particular solution of a recurrence relation. Under some special cases, however, we can make a reasonable guess of the particular solution based on the form of $f(n)$. This is summarised as follows

$f(n)$	$a_n^{(p)}$	
Ak^n	Bk^n	if k is not a characteristic root
	$Bn^m k^n$	if k is a characteristic root with multiplicity m
$\sum_{i=0}^{t} p_i n^i$	$\sum_{i=0}^{t} q_i n^i$	if 1 is not a characteristic root
	$n^m \sum_{i=0}^{t} p_i n^i$	if 1 is a characteristic root with multiplicity m
$An^t k^n$	$\left(\sum_{i=0}^{t} q_i n^i \right) k^n$	if k is not a characteristic root
	$n^m \left(\sum_{i=0}^{t} p_i n^i \right) k^n$	if k is a characteristic root with multiplicity m

Just like systems of equations, we can have two recurrence relations on sequences (a_n) and (b_n), which form a system of equations. We can solve the system by substitution to solve one of the recurrence relation first, just like solving a system of equations.

Note that for a sequence (a_n), we can define

$$A(x) = \sum_{i=0}^{\infty} a_i x^i$$

as the generating function. Recall that we can transform $A(x)$ to generate a linear combination of a_i's. Therefore, we could use generating functions to solve a recurrence relation.

3

Graph Theory

3.1   Graph Structures

Intuitively, we would describe a *graph* as a collection of nodes (or vertices) plus some lines (or edges) joining some nodes together. This can be rigorously defined as follows:

> **Definition 3.1.1 ▶ Multigraph**
>
> A **multigraph** G consists of a non-empty finite set of vertices denoted by $V(G)$ and a finite set of edges denoted by $E(G)$. $|V(G)|$ is known as the **order** of G, denoted by $v(G)$ and $|E(G)|$ is known as the **size** of G, denoted by $e(G)$.
>
> In particular, if $v(G) = m$ and $e(G) = n$, we say that G is an (m, n)-**graph**.
>
> G is said to be **trivial** if $v(G) = 1$ and **non-trivial** otherwise.

Note that in a multigraph, by default there can be a plural number of edges between any two vertices. We would define the notion of *simple graph* as a special multigraph.

> **Definition 3.1.2 ▶ Simple Graph**
>
> A multigraph G is said to be **simple** if there is at most one edge between any two distinct vertices.

> *Remark.* Note that if G is simple and undirected, then
>
> $$E(G) \subseteq \big\{ (v_j, v_i) : v_j, v_i \in V(G), j \geq i \big\}.$$

Notice that if (v_i, v_i) is an edge, it connects a vertex to itself. This is known as a *loop*. In a graph, it is important to know the layout of the vertices and edges. For this purpose, we define the notion of *adjacency* in a multigraph.

> **Definition 3.1.3 ▶ Adjacency and Neighbourhood**
>
> Let $v_i, v_j \in V(G)$, we say that they are **adjacent** if $v_i v_j \in E(G)$. Alternatively, we say that v_i and v_j are **neighbours** to each other. The edge $v_i v_j$ is said to be

incident with v_i and v_j. Two edges e and f are said to be **adjacent** if there exists some $v \in V(G)$ such that both e and f are incident with v.

The set of all neighbours to some $v_i \in V(G)$ is called the **neighbourhood set** of v_i, denoted by $N_G(v_i)$. In particular, the set $N_G[v_i] := N_G(v_i) \cup \{v_i\}$ is known as the **closed neighbourhood set** of v_i.

We can add on to the definition by discussing the size of the neighbourhood of a vertex.

Definition 3.1.4 ▶ Degree

The **degree** of v, denoted by $d_G(v)$ is defined as the number of edges incident to v. If $d_G(v)$ is even (repectively, odd), then we say that v is an even (respectively, odd) vertex. If $d_G(v) = 0$, we say that v is **isolated**; if $d_G(v) = 1$, we say that v is an **end** vertex.

In particular, we define

$$\Delta(G) = \max_{v \in G} d_G(v), \qquad \delta(G) = \min_{v \in G} d_G(v).$$

With the notion of degree established, we can now define a *regular graph*.

Definition 3.1.5 ▶ Regular Graph

A **regular graph** is a graph in which every vertex has the same degree. In particular, if $d_G(v) = k$ for all $v \in V(G)$, G is known as a k-**regular graph**.

Remark. A graph G is regular if and only if $\Delta(G) = \delta(G)$.

Since a graph essentially consists of two sets, it is natural to consider the notion of graph complementation. We now proceed to introducing the notion of *complement*.

Definition 3.1.6 ▶ Complement

Let G be a graph of order n, the **complement** of G, denoted by $\overline{G}$, is the graph of order n where
$$V\left(\overline{G}\right) = V(G), \qquad E\left(\overline{G}\right) = \{(u, v) : (u, v) \notin E(G)\}.$$

We discuss the following interesting question:

> 15 students went to a party. During the party some of them shook hands with each other. At the end of the party, the number of handshakes made by each

student was recorded and it was reported that the sum was 39. Was this possible?

Note that if we represent each student as a vertex, then we can use $V(G) = \{v_1, v_2, \cdots, v_{15}\}$ to construct a graph, in which an edge $v_i v_j \in E(G)$ if and only if students i and j shook hands. As such, $d_G(v_i)$ is the number of persons student i shook hands with.

Note that if we sum up $d_G(v_i)$ for all the vertices, every edge will be counted exactly twice! This means that

$$\sum_{i=1}^{15} d_G(v_i) = 39$$

is impossible since the left-hand side must be even. In fact, by the above reasoning, we see that in any graph, the sum of degrees of its vertices must be even.

Lemma 3.1.7 ▶ Handshaking Lemma

If G is a graph of order n and size m, then

$$\sum_{i=1}^{n} d_G(v_i) = 2m.$$

Remark. It can be easily deduced from the above lemma that in any graph, the number of vertices with odd degrees must be even.

As an extension of Lemma 3.1.7, the minimum size of any graph is obviously 0, and the maximum size of a simple graph occurs when there is an edge between any two vertices.

Definition 3.1.8 ▶ Empty and Complete Graphs

Let G be a simple graph of order n. G is said to be an **empty graph** or **null graph**, denoted by 0_n, if $e(G) = 0$, and a **complete graph**, denoted by K_n if for all $u, v \in V(G)$, we have $(u, v) \in E(G)$.

Since a multigraph is just two sets, we can define a "subset" relation between graphs.

Definition 3.1.9 ▶ Subgraph

Let G, H be graphs, then H is a **subgraph** of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. In particular, a subgraph H is a **proper subgraph** of G if $V(H) \neq V(G)$ or $E(H) \neq E(G)$.

Note that we a graph can possibly be reproduced by connecting its vertices correctly, so a subgraph containing all vertices of a graph "spans" the original graph.

> **Definition 3.1.10 ▶ Spanning Subgraph**
>
> Let H be a subgraph of G. H is called a **spanning subgraph** of G if $V(H) = V(G)$.

Note that by definition, a spanning subgraph retains all vertices of the original graph. Therefore, a way to quickly generate a subgraph of a given graph G is to keep the vertex set and delete some edges from the edge set. We denote such a graph by $H = G - F$ for some $F \subset E(G)$. Observe that

$$V(H) = V(G), \qquad E(H) = E(G) - F.$$

On the other hand, by deleting some vertices together with edges incident to them, we can produce a subgraph from any given graph.

> **Definition 3.1.11 ▶ Induced Subgraph**
>
> Let H be a subgraph of G. H is called a **induced subgraph** of G if
>
> $$E(H) = \{uv \in E(G) : u, v \in V(H)\}.$$
>
> Let $S \subseteq V(G)$, the subgraph of G induced by S is denoted by $[S]$.
>
> Alternatively, let $F \subseteq E(G)$. Define
>
> $$V' := \bigcup \{\{u, v\} : uv \in F\},$$
>
> then (V', F) is the subgraph of G induced by F, denoted by $G[F]$.

Intuitively, an induced subgraph pf G consists of a selected subset of the vertices of G together with all edges in G connecting any vertices in this subset.

Using the idea of deletion, we can quickly generate an induced subgraph H of G as follows: first, set $V(H) = V(G) - A$ for some $A \subseteq V(G)$, i.e., remove some vertices; then, we will remove all edges from $E(G)$ which are incident to some vertices in A, i.e., set

$$E(H) = \{e \in E(G) : e \text{ is not incident to any } v \in A\}.$$

Then, $H = \big(V(H), E(H)\big)$ is an induced subgraph of G. In fact, we can denote H as

$$H = G - A = [V(G) - A].$$

This leads to the following proposition:

> **Proposition 3.1.12 ▶ Characterisation of Induced Subgraphs**
>
> *Let G be a graph. If $A \subseteq V(G)$, then $G - A = [V(G) - A]$. Suppose H is a subgraph of G, then H is an induced subgraph of G if and only if $H = G - (V(G) - V(H))$.*
>
> *Proof.* Left to the reader as an exercise.   □

3.2   Graph Isomorphism

Given two graphs G and H, are they the same graph? This seemingly innocent question proves to be extremely hard to answer. Two graphs can look drastically different but be structurally identical in reality. For example, we could shift around the vertices of a graph without changing any edge to alter the shape of the graph dramatically. Therefore, to compare the structures of graphs, we require some rigorous definition.

> **Definition 3.2.1 ▶ Graph Isomorphism**
>
> Two graphs G and H are said to be **isomorphic**, denoted by $G \cong H$, if there exists a bijection $f : V(G) \to V(H)$ such that
>
> $$uv \in E(G) \quad \text{if and only if } f(u)f(v) \in E(H).$$

Such a bijection f is said to **preserve adjacency**, i.e., if u and v are neighbours in G, then their images are also neighbours in H.

It is very hard to determine whether two specific graphs are isomorphic, but there are some considerations in the general case. For example, some trivial conclusions include:

- Two graphs with different orders cannot be isomorphic.

- Two graphs with different sizes cannot be isomorphic.

- Two graphs with different numbers of components cannot be isomorphic.

- If the numbers of vertices with degree k are different in two graphs, they cannot be isomorphic.

Notice that by Definition 3.2.1, we can define a function g such that $uv \notin E(G)$ if and only if $g(u)g(v) \notin E(H)$ and relate this function to the complement graphs of G and H. Here we introduce a way to determine isomorphism by considering complement graphs.

Theorem 3.2.2 ▶ Complementation Preserves Isomorphism

Let G and H be two graphs of the same order. $G \cong H$ if and only if $\overline{G} \cong \overline{H}$.

Proof. Left to the reader as an exercise.   □

Now let us think the reverse: if it is not easy to prove isomorphism, can we find a way to quickly determine that two graphs are not isomorphic? Here we present the necessary conditions for isomorphism:

Theorem 3.2.3 ▶ Necessary Conditions for Isomorphism

If $G \cong H$, then
1. *G and H must have the same order and size.*
2. *$\delta(G) = \delta(H)$ and $\Delta(G) = \Delta(H)$.*
3. *The number of vertices with degree i in G and H is the same for all $i \in \mathbb{N}$.*

The first two conditions are easy to observe. For the third condition, we introduce a tool known as *degree sequences*.

Definition 3.2.4 ▶ Degree Sequence

Let G be a graph of order n. If we label its vertices by $v_1, v_2, \cdots, v_n$ such that

$$d(v_1) \geq d(v_2) \geq \cdots \geq d(v_n),$$

then the non-increasing sequence $(d(v_n))$ is known as the **degree sequence** of G.

Now we consider the following question:

> Let (d_n) be a non-increasing sequence, is there some graph whose degree sequence is (d_n)?

We will make use of the following definition:

Definition 3.2.5 ▶ Graphic Sequence

Let (d_n) be a sequence of non-negative integers at most $n-1$. (d_n) is said to be **graphic** if there exists a graph G whose degree sequence is (d_n).

To determine whether a sequence is graphic by eye power is difficult. Fortunately, we have the following recursive approach:

Theorem 3.2.6 ▸ Havel-Hakimi Algorithm

Let (d_n) be a non-increasing sequence of non-negative integers at most $n - 1$. Define

$$
d_m^* = \begin{cases} d_{m+1} - 1 & \text{if } 1 \leq m \leq d_1 \\ d_{m+1} & \text{if } d_1 + 1 \leq m \leq n - 1 \end{cases}.
$$

(d_n) is graphic if and only if (d_m^*) is graphic.

Proof. Suppose (d_n) is graphic, we consider the following lemma:

Lemma 3.2.7

For any graphic sequence (d_n), there is some graph G of order n with $d_G(v_i) = d_i$ for $i = 1, 2, \cdots, n$ such that v_1 is adjacent to v_j for $j = 2, 3, \cdots, d_1 + 1$.

Proof. Suppose on contrary there is no such a graph G, then for any graph G with degree sequence (d_n), v_1 is adjacent to at most $d_1 - 1$ vertices in

$$
A := \{v_j : j = 2, 3, \cdots, d_1 + 1\}.
$$

Thus, there exists some $v_j \in A$ such that $v_1 v_j \notin E(G)$. However, $d_G(v_1) = d_1$, so there exists some $v_k \in V(G) - A$ with $v_k \neq v_1$ such that $v_1 v_k \in E(G)$.

Notice that since $v_k \notin A \cup \{v_1\}$, $k > j$ and so $d_G(v_k) = d_k \leq d_j = d_G(v_j)$. Therefore, v_j has at least as many neighbours as v_k, which means there must exists some v_r with $v_j v_r \in E(G)$ such that $v_k v_r \notin E(G)$.

Define a graph G' by

$$
V(G') = V(G), \qquad E(G') = E(G) - v_j v_r - v_1 v_k + v_j v_k + v_1 v_j.
$$

Note that $d_{G'}(v_i) = d_G(v_i)$ for all $i = 1, 2, \cdots, n$, so (d_n) is also a degree sequence for G'. However, now v_1 in G' is adjacent to all vertices in A, which is a contradiction. $\square$

By Lemma 3.2.7, we can choose some graph G whose degree sequence is (d_n) such that v_1 is adjacent to d_1 vertices $v_2, v_3, \cdots, v_{d_1+1}$. Let $H = G - v_1$, then for any $u_i \in H$,

$$
d_H(u_i) = \begin{cases} d_G(v_{i+1}) - 1 & \text{if } 1 \leq i \leq d_1 \\ d_G(v_{i+1}) & \text{if } d_1 + 1 \leq i \leq n - 1 \end{cases}.
$$

Therefore, (d_m^*) is graphic.

Conversely, suppose (d_m^*) is graphic, then there is some graph H of order $n - 1$ such that $d_H(u_i) = d_i^*$ for $i = 1, 2, \cdots, d_1$. Construct a graph $G = H + v_1$ such that $v_i = u_{i-1}$ for $i = 2, 3, \cdots, n$ and v_1 is adjacent to $v_2, v_3, \cdots, v_{d_1+1}$. Therefore,

$$
d_G(v_i) = \begin{cases} d_1 & \text{if } i = 1 \\ d_{i-1}^* + 1 & \text{if } 2 \leq i \leq d_1 + 1 \\ d_i^* & \text{if } d_1 + 2 \leq i \leq n \end{cases} \cdot
$$

Note that $d_G(v_i) = d_i$, so (d_n) is graphic. $\qquad \square$

By Theorem 3.2.6, we can reduce a sequence recursively until we reach some obvious case. This obvious case is graphic if and only if the original sequence is graphic. We can also build the graph corresponding to the original sequence from a simple graphic sequence.

Lastly, we introduce a special type of graphs known as *self-complementary* graphs.

Definition 3.2.8 ▸ Self-Complementary Graph

A graph G is said to be **self-complementary** if $G \cong \overline{G}$.

It turns out that a self-complementary graph satisfies some special properties.

Proposition 3.2.9 ▸ Order of Self-Complementray Graphs

If G is a self-complementary graph of order n, then either $n = 4k$ or $n = 4k + 1$ for some $k \in \mathbb{Z}^+$.

Proof. Since G is self-complementary, $G \cong \overline{G}$ and so $e(G) = e\left(\overline{G}\right)$. Note that

$$
e(G) + e\left(\overline{G}\right) = \binom{n}{2} = \frac{n(n-1)}{2},
$$

so we have $e(G) = \frac{n(n-1)}{4}$. Since $e(G) \in \mathbb{Z}^+$, either n or $n-1$ is divisible by 4, so $n = 4k$ or $n = 4k + 1$ for some $k \in \mathbb{Z}^+$. $\qquad \square$

3.3   Graph Traversal

Next, we study the movement within a graph, that is, how to traverse the vertices in a graph along its edges.

> **Definition 3.3.1 ▶ Walk, Trail, Path**
>
> Let G be a graph and $x, y \in V(G)$, then an *x-y* **walk** is an alternating sequence
>
> $$W := v_0 e_1 v_1 e_2 v_2 \cdots v_{n-1} e_n v_n$$
>
> where $v_i \in V(G)$, $e_i = v_{i-1} v_i \in E(G)$, $x = v_0$ and $y = v_n$. In particular, W is **open** if $v_0 \neq v_n$ and **closed** otherwise. The **length** of a walk is defined as the number of edges in it.
>
> If $e_i \neq e_j$ whenever $i \neq j$, then W is called a **trail**. A closed trail is called a **circuit**. If $v_i \neq v_j$ whenever $i \neq j$, then W is called a **path**. A closed path is called a **cycle**.

> *Remark.* All paths are trails and all trails are walks, but the converses are not true.

In more informal terms, a trail is a walk with no repeated edge and a path is a trail with no repeated vertex.

Note that a path is essentially a gragh. We denote a path of order n by P_n and a cycle of order n by C_n.

A common tool in proofs of graph theory is the **extremal approach**, which can be demonstrated in the proposition below.

> **Proposition 3.3.2 ▶ Reduction of Walks to Paths**
>
> *If a graph G conatins a u-v walk of length k, then G contains a u-v path of length at most k.*
>
> ---
>
> *Proof.* Let S be the set of all u-v walks in G, Note that $S \neq \varnothing$ since there is a u-v walk of length k. Let P be a walk of the shortest length. We claim that P must be a path. Suppose P is not a path, then there exists some vertices $w_1 = w_2$ in P. Suppose Q is obtained by removing the w_1-w_2 walk from P, then Q is a walk. However, Q is shorter than P which is not possible. Therefore, P must be a path. Note that the length of P is at most k, so G contains a u-v path of length at most k. $\qquad\square$

An important question we are interested in with graphs is the reachability of a vertex, i.e., given two vertices u and v, we want to know whether we can reach one vertex from another. Intuitively, we can traverse between two vertices if there is a path between them. From here we define the notion of a *connected graph*.

> **Definition 3.3.3 ▶ Connected Graph**
>
> Let G be a graph. Two vertices u and v are **connected** if there is a path betweem them. G is said to be **connected** if for any $u, v \in V(G)$, there exists a path from u to v. For any $u \in V(G)$, we denote the set of all vertices connected to u (inclusive of u) as $c(u)$.

Note that connectedness is an equivalence relation. Let P be a partition formed by the equivalence classes of G under the connectedness relation, then for any $u, v \in V(G)$, u and v are in the same equivalence class if and only if u and v are connected. Any $c(u) \in P$ induces a subgraph $[c(u)]$.

> **Definition 3.3.4 ▶ Connected Component**
>
> Let G be a graph and $\sim$ be a relation such that for any $u, v \in V(G)$, $u \sim v$ if and only if u and v are connected. Let
> $$C := V(G)/\sim$$
> be the quotient set, then any $c(u) \in C$ is known as a (connected) **component** of G. The number of components of G is denoted by $\omega(G)$.

It can be easily seen that G is connected if and only if $\omega(G) = 1$. We will now establish a relationship between connectedness and complementation.

> **Theorem 3.3.5 ▶ Connectedness of Complement**
>
> *If G is disconnected, then $\overline{G}$ is connected.*
>
> ---
>
> *Proof.* Let $u, v \in V\left(\overline{G}\right)$ be two arbitrary vertices. If $uv \notin E(G)$, then $uv \in \left(\overline{G}\right)$ and so u and v are connected in $\overline{G}$. If $uv \in E(G)$, since G is disconnected, there exists some $w \in V(G)$ such that $uw, wv \notin E(G)$. Therefore, $uw, wv \in E\left(\overline{G}\right)$. This means that there is a u-v path in $\overline{G}$ and so u and v are connected in $\overline{G}$. Therefore, $\overline{G}$ is connected. $\qquad\square$

Intuitively, we can transform a connected graph into a disconnected one by deleting some vertices or edges. It can be easily observed that in certain graphs, deleting one particular vertex or edge will immediately disconnect the graph.

> **Definition 3.3.6 ▶ Cut-Vertex, Bridge**
>
> Let G be a non-trivial graph. A vertex $v \in V(G)$ is a **cut-vertex** if $\omega(G - v) > \omega(G)$. An edge $e \in E(G)$ is a **bridge** if $\omega(G - e) > \omega(G)$.

Note that a graph does not have to be connected to have cut vertices and bridges. Essen-

tially, a cut-vertex (or bridge) is just a vertex (or edge) which disconnects a component upon removal.

Now the next question is how do we identify a cut-vertex or a bridge in a graph? Intuitively, a cut-vertex divides a graph into two portions such that traversal between the two portions has to pass through it.

Theorem 3.3.7 ▶ Cut-Vertex Characterisation

Let G be a graph. $v \in V(G)$ is a cut-vertex if and only if there exists $a, b \in V(G)$ such that v is in every path between a and b.

Proof. Let $v \in V(G)$ be a cut-vertex and consider $H = G - v$, then we can find two non-empty disjoint connected components of H, say H_1 and H_2. Take $a \in V(H_1)$ and $b \in V(H_2)$, then they are disconnected. However, a and b are connected in G, which means v is in every a-b path.

We will prove the converse by contrapositive. Suppose v is not a cut-vertex of G, then $G - v$ is connected. Therefore, for any $u, w \in V(G)$, there is a path between them which does not contain v. $\qquad\square$

A similar argument can be established for bridges. Furthermore, since we have to pass through the bridge when traversing between the two portions, it can be easily seen that we have to reuse the bridge in order to traverse back.

Theorem 3.3.8 ▶ Bridge Characterisation

Let G be a graph. $e \in E(G)$ is a bridge if and only if e is not contained by any cycle in G.

Proof. We will prove the forward direction by contrapositive. Suppose $e = xy$ is contained in some cycle in G, then there is some x-y path in $G - e$. Suppose $a, b \in V(G)$ are connected in G via a path containing e, this implies that there is some a-b path in $G - e$ and so $G - e$ is connected, which means that e is not a bridge.

We will prove the converse also by contrapositive. Suppose $e = xy$ is not a bridge, the $G - e$ is connected and so there is some x-y path in $G - e$. Let this path be P, then $P \cup \{e\}$ is a cycle in G. $\qquad\square$

Combining the two characterisations, we can relate cut-vertices to bridges in the following proposition:

> **Proposition 3.3.9 ▶ Bridge Incident to A Cut-Vertex**
>
> *Let G be a graph. If $uv \in E(G)$ is a bridge and u is not an end vertex, then u is a cut-vertex.*
>
> *Proof.* Left to the reader as an exercise. □

A direct consequence of this is the following corollary:

> **Corollary 3.3.10 ▶ Co-existence of Bridges and Cut-Vertices**
>
> *If G is a graph with order at least 3 and G contains a bridge, then G contains a cut-vertex.*

3.4   Weighted Graphs

Given a graph G with u, v being two connected vertices, we are interested in the notion of distance between them. Intuitively, if there are multiple paths between u and v, we would take the length of the shortest one to represent their distance.

> **Definition 3.4.1 ▶ Distance, Eccentricity, Diameter**
>
> Let G be a connected graph and let $u, v \in V(G)$. The **distance** between u and v, denoted by $d(u, v)$, is the length of the shortest path between u and v. The **eccentricity** of u is defined to be
> $$e(u) = \max_{v \in V(G)} \{d(u, v)\}.$$
>
> The **diameter** of G is defined by
> $$\mathrm{diam}(G) = \max_{u \in V(G)} \{e(u)\}.$$
>
> The **radius** of G is defined by
> $$\mathrm{rad}(G) = \min_{u \in V(G)} \{e(u)\}.$$
>
> A vertex v is called a **central** vertex if
> $$e(v) = \mathrm{rad}(v).$$
>
> The subgraph induced by the set of central vertices of G is known as the **centre** of G.

This also justifies the use of words "diameter" and "radius" in a circle. We can view a circle as a graph consisting of a centre and infinitely many vertices in the circumference, with

edges of length r connecting the centre to the circumference. Indeed, in this definition, the farthest vertices will be any two on the circumference with a distance of $2r$, and the nearest vertices will be the centre and any vertex on the circumference with a distance of r.

For readers with knowledge in real analysis or topology, it is easy to see that an undirected unweighted connected graph is a metric space with distance between vertices as its metric. Thus it is natural to have the following result:

Theorem 3.4.2 ▸ Triangle Inequality

If G is connected, then for any $u, v, w \in V(G)$,

$$d(u, v) \le d(u, w) + d(w, v).$$

Proof. Let U, V be the shortest u-w path and the shortest w-v path respectively, then $U + V$ is a u-v walk with length $d(u, w) + d(w, v)$. Since the shortest u-v path has length $d(u, v)$, we have

$$d(u, v) \le d(u, w) + d(w, v).$$

$\square$

An application of Theorem 3.4.2 allows us to establish the following:

Theorem 3.4.3 ▸ Boundedness of Diameter

If G is connected, then
$$\mathrm{rad}(G) \le \mathrm{diam}(G) \le 2\mathrm{rad}(G).$$

Proof. $\mathrm{rad}(G) \le \mathrm{diam}(G)$ is immediate from Definition 3.4.1. Take $u, v \in V(G)$ with $d(u, v) = \mathrm{diam}(G)$ and take $w \in V(G)$ such that $e(w) = \mathrm{rad}(G)$. Notice that this implies

$$d(u, w) \le e(w) = \mathrm{rad}(G)$$
$$d(w, v) \le e(w) = \mathrm{rad}(G).$$

By Theorem 3.4.2,

$$\mathrm{diam}(G) = d(u, v) \le d(u, w) + d(w, v) = 2\mathrm{rad}(G).$$

$\square$

The notion of distance between vertices can be abstracted into a concept known as edge (vertex) weight. So far we have been dealing with graphs in which edges do not differ from

one another except the vertices they are incident to. However, in real life, there are many scenarios where some egdes in a graph are "more important" than others. For example, when building roads to connect villages, we wish to choose edges with the lowest cost. Therefore, we introduce the notion of *weighted graphs*

> ### Definition 3.4.4 ▶ Weighted Graph
>
> A **weighted graph** is a graph G with a function $w \colon E(G) \to \mathbb{R}$. For any $e \in E(G)$, $w(e)$ is called the **weight** of e. If H is a subgraph of G, the **weight** H is defined to be
> $$\sum_{e \in E(H)} w(e).$$

One classic problem on weighted graphs is the (single-source) shortest path problem.

In an unweighted graph, we define the length of a path to be the number of edges it contains. However, in a weighted graph we need a different definition to distinguish edges of different weights.

> ### Definition 3.4.5 ▶ Length of a Weighted Path
>
> Let G be a weighted graph and P be a path in G. The **length** of P is defined as $w(P)$.

Similarly, we can define the shortest path in a weighted graph as the path with the lowest weight, and the distance between two vertex as the weight of the shortest path between them. The (single-source) shortest path problem asks:

> Let G be a weighted graph and s be the source vertex. For any $v \in V(G)$, what is the distance between s and v?

Let us consider the following idea: suppose we are already certain that the distance from s to u cannot be further reduced, then we can safely take the shortest edge from u to u' and compute the weight of the s-u' path containing u. If this path is shorter than the known distance from s to u', we will "relax" this distance. A common and famous algorithm to solve a (single-source) shortest path problem is the *Dijkstra's algorithm*.

> ### Technique 3.4.6 ▶ Dijkstra's Algorithm
>
> Let G be a weighted graph and u_0 be the source.
> - Define $L \colon V(G) \to \mathbb{R}$: A function which maps a vertex in G to the current distance between it and the source vertex.
> - Initialise $L(u_0) = 0$ and $L(u_i) = \infty$ for all other vertices.
> - Declare all vertices as temporary.

- Let v be the vertex such that $L(v)$ is the lowest among all temporary vertices. Declare v as final.
- For each neighbour u_i of v:
  - If $w(vu_i) + L(v) < L(u_i)$, relax $L(u_i)$.
- Repeat the process until all vertices have become final.

In real programming, the notion of "temporary" and "final" vertices can be easily implemented with a priority queue to achieve logarithmic complexity in distance update.

3.5 Graphs and Matrices

So far we have been using hand-drawn illustrations to aid us in graph visualisation. However, for large graphs with many vertices and egdes, drawing out the graph structures becomes nearly impossible. Therefore, we need to use some algebraic tools to help us represent a large graph.

Definition 3.5.1 ▶ Adjacency Matrix

Let G be a multigraph with $v(G) = n$, then the **adjacency matrix** denoted by $A(G)$ is an $n \times n$ matrix such that a_{ij} is the number of edges incident to both $v_i, v_j \in V(G)$.

Remark. For an undirected multigraph, A is always symmetric.

Note that an adjacency matrix does not tell us which edges are incident to a particular vertex, so we may consider the following alternative:

Definition 3.5.2 ▶ Incidence Matrix

Let G be a multigraph with $v(G) = n$ and $e(G) = m$, then the **incidence matrix** denoted by $M(G)$ is an $n \times m$ matrix such that

$$m_{ij} = \begin{cases} 1, & \text{if } e_j \text{ is incident with } v_i \\ 0, & \text{otherwise} \end{cases}.$$

Remark. For an undirected multigraph, consider an arbitrary k-th column of M. Note that any column will have exactly 2 non-zero entries. Suppose $m_{ik} = m_{jk} = 1$, it means that $e_k = v_i v_j$.

For readers who have learnt about basic data structures in computer science, an incidence matrix is just an adjacency list with each list of neighbours converted to a (non-compact)

boolean array. Note that the sum of the i-th row for both A and M is the degree of vertex i. Since A and M carries information about adjacencies between vertices, we can use them to draw some conclusions regarding graph traversal.

Proposition 3.5.3 ▶ Number of Walks with Given Length

Let G be a simple graph of order n with adjacency matrix A, then the (i, j) entry of A^k is the number of v_i-v_j walks of length k.

Proof. The case where $k = 1$ is trivially true since there is a v_i-v_j walk of length 1 if and only if v_i and v_j are neighbours. Let the (i, j) entry of A^m be m_{ij}. Suppose m_{ij} is the number of v_i-v_j walks of length m. Then the (i, j) entry of A^{m+1} is given by

$$\sum_{r=1}^{n} m_{ir} a_{rj}.$$

Notice that m_{ir} and a_{rj} are the number of v_i-v_r walks of length m and the number of v_r-v_j walks of length 1 respectively, so by Theorem 1.1.2, $m_{ir} a_{rj}$ is the number of v_i-v_j walks of length n that passes through v_r. Therefore, summing up $m_{ir} a_{rj}$ for each $r = 1, 2, \cdots, n$ gives the total number of v_i-v_j walks of length $n + 1$. $\qquad\square$

Additionally, we have the following corollary:

Corollary 3.5.4 ▶ Matrix Characterisation of Connected Graphs

Let G be a graph with $v(G) \geq 2$ and adjacency matrix A. Let

$$B = \sum_{i=1}^{n-1} A^i,$$

then G is connected if and only if $b_{ij} \neq 0$ for all $i \neq j$.

3.6   Bipartite Graphs

Definition 3.6.1 ▶ Bipartite Graph

A graph G is **bipartite** if $V(G)$ can be partitioned into two disjointed subsets V_1 and V_2 such that every edge of G joins a vertex in V_1 to a vertex in V_2. (V_1, V_2) is known as a **bipartition** of G.

Note that the size of a bipartite graph G with partite sets V_1 and V_2 is just

$$e(G) = \sum_{x \in V_1} d_G(x) = \sum_{y \in V_2} d_G(y).$$

It can be deduced that the maximal size of a bipartite graph is achieved when every vertex in V_1 is connected to every vertex in V_2. To define this rigorously, we first introduce the notion of *join*.

Definition 3.6.2 ▶ Join

Let G_1 and G_2 be two disjoint graphs of order n_1 and n_2 respectively. The **join** of G_1 and G_2, deonoted by $G_1 + G_2$ is the graph such that

$$V(G_1 + G_2) = V(G_1) \cup V(G_2)$$
$$E(G_1 + G_2) = E(G_1) \cup E(G_2) \cup \{uv : u \in V(G_1), v \in V(G_2)\}.$$

Informally, we can describe the join of G_1 and G_2 as a graph which contains all the vertices from G_1 and G_2, all existing edges from G_1 and G_2, and new edges connecting every vertex in G_1 to every vertex in G_2. Thus, we can define a complete bipartite graph using a join.

Definition 3.6.3 ▶ Complete Bipartite Graph

A **complete bipartite graph** is defined as $K_{p,q} = 0_p + 0_q$.

Sometimes, it is not easy to identify the bipartition in a bipartite graph. Therefore, to quickly determine whether a graph is bipartite, we consider the following theorem:

Theorem 3.6.4 ▶ Bipartite Graph Characterisation

A graph is bipartite if and only if it contains no odd cycles.

Proof. Suppose G is bipartite with bipartition (U, V). If G contains no cycle then we are done. Suppose G contains some cycle. Let u_i be the i-th visited vertex from U in the cycle. Note that between u_i and u_{i+1} there exists one and only one vertex from V. Suppose k vertices from U are contained in the cycle, then it is easy to see that there are also k vertices from V contained in the cycle. Therefore, every cycle in G must be even.

Suppose conversely that G contains no odd cycles. Note that it suffices to prove that every component of G is bipartite. Let C be a component of G, then C contains no

---

odd cycles. Take some $w \in V(C)$ and define the sets

$$V_1 := \{v : d(v, w) \text{ is even}\}$$
$$V_2 := \{v : d(v, w) \text{ is odd}\}.$$

We will prove (V_1, V_2) is a bipartition. Suppose it is not a bipartition, then without loss of generality, let $a, b \in V_1$ be neighbours. Note that there is a path A between w and a of even length and a path B between w and b of even length, so $A \cup B \cup \{ab\}$ is a closed walk of odd length. Consider the following lemma:

Lemma 3.6.5

A closed walk of odd length in a graph contains an odd cycle.

Proof. Let the walk have a length of p. The case where $p = 3$ is trivially true since the only closed walk of length 3 is C_3.

Suppose any closed walk of odd length less than p contains an odd cycle. Let P be a closed walk of length p. If P is a cycle then we are done. If P is not a cycle, then there exist some $w_1 = w_2$ in P. Note that $w_1 = w_2$ is a cut-vertex. So there exists two closed walks Q_1 and Q_2 such that $Q_1 \cap Q_2 = \{w_1\}$. Notice that one of Q_1 and Q_2 must have an odd length and so contains an odd cycle, so P contains an odd cycle. □

By Lemma 3.6.5, $A \cup B \cup \{ab\}$ contains an odd cycle, which is a contradiction. □

3.7   Trees

Definition 3.7.1 ▶ Tree

A **tree** is a **connected** graph with **no cycles**.

Consider the following question:

> What is the minimal number of edges needed to construct a connected graph of order n?

Intuitively, we need at least $(n - 1)$ edges. On the other hand, it is also easy to see that a tree of order n is of the smallest size among all order-n graphs.

> ### Theorem 3.7.2 ▶ Equivalent Definitions for Trees
>
> *Let T be a connected graph of order n and size m, then the following statements are equivalent:*
>
> 1. *T is a tree;*
> 2. *Every two vertices in T are joined by a unique path;*
> 3. *$m = n - 1$.*
>
> *Proof.* Left to the reader as an exercise. □

We introduce 2 interesting properties of trees.

> ### Proposition 3.7.3 ▶ Number of Vertices with Given Degree in a Tree
>
> *Let T be a tree with $\Delta(T) = k$. Let n_i be the number of vertices in T of degree i, then*
>
> $$n_1 = 2 + \sum_{i=1}^{k-2} i n_{i+2}.$$
>
> *Proof.* Let the order of T be n, then $n = \sum_{i=1}^{k} n_i$. By Lemma 3.1.7, we have
>
> $$\sum_{v \in V(T)} d(v) = 2(n-1).$$
>
> However, since there are n_i vertices with degree i, we have
>
> $$\sum_{v \in V(T)} d(v) = \sum_{i=1}^{k} i n_i.$$
>
> Therefore,
>
> $$2 \sum_{i=1}^{k} n_i - 2 = \sum_{i=1}^{k} i n_i$$
>
> $$n_1 = \sum_{i=3}^{k} i n_i$$
>
> $$n_1 = \sum_{i=1}^{k-2} i n_{i+2}.$$
>
> □

The following corollary is an immediate consequence of Proposition 3.7.3:

> **Corollary 3.7.4 ▶ Every Non-trivial Tree Has at Least 2 Leaves**
>
> *A tree of order at least 2 contains at least 2 leaves.*

A tree with exactly 2 end vertices is a doubly linked list and a collection of disjoint trees are vividly named as a *forest*.

> **Definition 3.7.5 ▶ Forest**
>
> A **forest** is a graph G in which each component is a tree.

3.7.1   Spanning Trees

A *spanning tree* is a special type of tree obtained as a subgraph containing all vertices from the original graph.

> **Definition 3.7.6 ▶ Spanning Tree**
>
> Let G be a multigraph, a tree T which is a spanning graph of G is known as a **spanning tree** of G.

Intuitively, one can traverse along a spanning tree of a graph to reach every vertex of the graph from any source vertex. Therefore, a spanning tree essentially gives all paths between any pair of vertices, and so the existence of a spanning tree is an indicator of connectedness.

> **Proposition 3.7.7 ▶ Spanning Tree Characterisation of Connected Graphs**
>
> *A graph G is connected if and only if it contains a spanning tree.*
>
> *Proof.* If G contains a spanning tree, then every pair of vertices in G are connected, and so G is connected. Conversely, suppose that G is connected. If G is a tree, then it is a spanning tree of itself. Otherwise, G contains some cycle C. Let e be any edge in C, then e is not a bridge so $G-e$ is connected. Set $G' = G-e$ and repeat the process for G'. Since G is finite, we will eventually remove all cycles in G and derive a tree containing all vertices in G, i.e., a spanning tree. □

Notice that a connected graph cannot be smaller in size than its spanning tree, so we have the following corollary:

> **Corollary 3.7.8 ▶ Minimal Size of Connected Graphs**
>
> *If G is a connected graph of order n, then $e(G) \geq n - 1$.*

Now, we wish to answer the following question:

> Let G be a connected multigraph of order n and size m. Suppose the vertices of G are labelled and duplicated egdes between any pair of vertices distinguished, how many different spanning trees of G there are?

Let G be a connected multigraph, we denote **the number of spanning trees** of G by $\tau(G)$. It is not an easy task to count directly by observation. We introduce a useful concept known as *contraction*.

Definition 3.7.9 ▶ Contraction

Let G be a connected multigraph of order n and let $e = uv$ be an egde of G. A **contraction** of G by e, denoted by $G \circ e$, is produced by removing all edges joining u and v and identifying u and v into a single vertex.

We may make use of contractions to compute $\tau(G)$ in a recursive manner.

Theorem 3.7.10 ▶ Recursive Algorithm for $\tau(G)$

Let G be a connected multigraph, then

$$\tau(G) = \tau(G - e) + \tau(G \circ e)$$

for all $e \in E(G)$.

Proof. Let G be a multigraph and $e \in E(G)$. Define A to be the set of spanning trees of G which contains e and B to be that which does not contain e. Take any $T_A \in A$, then $e \in E(T_A)$ is a bridge in T_A. Notice that if $T_1, T_2 \in A$, then $T_1 \neq T_2$, so they have at least one different edge which is not e. Define

$$A' := \left\{ T_A' = T_A - e : T_A \in A \right\}.$$

Clearly, for $T_1', T_2' \in A'$, $T_1' \neq T_2'$ are different spanning trees. Observe that there is a bijection between A and A', so $|A| = |A'| = \tau(G \circ e)$. Take any $T_B \in B$, since $e \notin E(T_B)$, this means that T_B must be a spanning tree of $G - e$. Therefore, $|B| = \tau(G - e)$. Hence,

$$\tau(G) = |B| + |A| = \tau(G - e) + \tau(G \circ e).$$

$\square$

Theorem 3.7.10 allows us to reduce $\tau(G)$ for a complex graph G into a linear combination of τ's for simpler graphs. So now we will discuss some basic cases where τ is easy to compute.

1. A simple cycle of order n:

   Note that for any $e \in E(C_n)$, $C_n - e$ is a spanning tree, so $\tau(C_n) = e(C_n) = n$.

2. A connected graph G with a cut-vertex v:

   Let G_1 and G_2 be two components of $G - v$. A spanning tree of G consists of a spanning tree of $[V(G_1) \cup \{v\}]$ and a spanning tree of $[V(G_2) \cup \{v\}]$, so

   $$\tau(G) = \tau\big([V(G_1) \cup \{v\}]\big)\tau\big([V(G_2) \cup \{v\}]\big).$$

3. A connected graph G with a bridge e:

   By a similar argument as the previous case, let G_1 and G_2 be the two components of $G - e$, then $\tau(G) = \tau(G_1)\tau(G_2)$.

4. A connected graph G consisting of two simple cycles C_p and C_q sharing a common edge e:

   Note that $G - e$ is a simple cycle C_{p+q-2} and $G \circ e$ is two simple cycles C_{p-1} and C_{q-1} sharing a cut-vertex. By Theorem 3.7.10,

   $$\begin{aligned} \tau(G) &= \tau(G - e) + \tau(G \circ e) \\ &= p + q - 2 + (p-1)(q-1). \end{aligned}$$

5. A simple cycle C_p with exactly one edge uv duplicated.

   Let e and f be the duplicated edges. Observe that there is only 1 spanning tree which contains neither e nor f. If exactly one of e and f is part of the spanning tree, that spanning tree also spans a simple cycle C_p. Therefore, $\tau(G) = 2(p-1) + 1 = 2p - 1$.

6. A connected graph G consisting of two simple cycles C_p and C_q sharing a pair of duplicated edges e and f:

   Note that $G - e$ is just the graph in case 4 and $G \circ e$ is C_{p-1} and C_{q-1} joined by a common cut-vertex, so by Theorem 3.7.10,

   $$\begin{aligned} \tau(G) &= \tau(G - e) + \tau(G \circ e) \\ &= p + q - 2 + 2(p-1)(q-1). \end{aligned}$$

When dealing with large graphs, we can make use of matrices just like what we did for graph traversal. In particular, we shall state the following useful theorem without proof.

Theorem 3.7.11 ▸ Matrix Tree Theorem

Let A be the adjacency matrix of a multigraph G and C be the $n \times n$ diagonal matrix defined by

$$c_{ij} = \begin{cases} d(v_i) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}.$$

Then, $\tau(G)$ is the cofactor of any entry of $C - A$.

Using Theorem 3.7.11, we can prove the following propositions:

Proposition 3.7.12 ▸ Number of Spanning Trees for Complete Graphs

Let $n \in \mathbb{N}$ and $n \geq 2$ and let K_n be a complete graph, then $\tau(K_n) = n^{n-2}$.

Proof. Note that since K_n is complete, $C(K_n)$ is defined by

$$c_{ij} = \begin{cases} n - 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

and the adjacency matrix $A(K_n)$ is such that

$$a_{ij} = \begin{cases} 1 & \text{if } i \neq j \\ 0 & \text{otherwise} \end{cases}.$$

Therefore, the cofactor of $C(K_n) - A(K_n)$ will be the $(n-1) \times (n-1)$ determinant

$$\begin{vmatrix} n-1 & -1 & -1 & \cdots & -1 \\ -1 & n-1 & -1 & \cdots & -1 \\ -1 & -1 & n-1 & \cdots & -1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & -1 & \cdots & n-1 \end{vmatrix} \longrightarrow \begin{vmatrix} 1 & 1 & 1 & \cdots & 1 \\ -1 & n-1 & -1 & \cdots & -1 \\ -1 & -1 & n-1 & \cdots & -1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & -1 & \cdots & n-1 \end{vmatrix},$$

where the latter is obtained by adding all other rows to the first row. Performing row

operations by adding the first row to every other row, the determinant becomes

$$
\begin{vmatrix}
1 & 1 & 1 & \cdots & 1 \\
0 & n & 0 & \cdots & 0 \\
0 & 0 & n & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & n
\end{vmatrix} = n^{n-2}.
$$

Since such row operations do not change the determinant, $\tau(K_n) = n^{n-2}$. $\square$

Proposition 3.7.13 ▶ Number of Spanning Trees for Complete Bipartite Graphs

For all $r \in \mathbb{Z}^+$, let $K_{2,r}$ be a complete bipartite graph with partite sets of sizes 2 and r, then $\tau\left(K_{2,r}\right) = 2^{r-1}r$.

Proof. Note that

$$
C(K_{2,r}) - A(K_{2,r}) =
\begin{bmatrix}
r & 0 & -1 & -1 & \cdots & -1 \\
0 & r & -1 & -1 & \cdots & -1 \\
-1 & -1 & 2 & 0 & \cdots & 0 \\
-1 & -1 & 0 & 2 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & 0 & 0 & \cdots & 2
\end{bmatrix}.
$$

Consider the $(r+2, r+2)$ cofactor:

$$
\begin{vmatrix}
r & 0 & -1 & -1 & \cdots & -1 \\
0 & r & -1 & -1 & \cdots & -1 \\
-1 & -1 & 2 & 0 & \cdots & 0 \\
-1 & -1 & 0 & 2 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & 0 & 0 & \cdots & 2
\end{vmatrix}
\longrightarrow
\begin{vmatrix}
1 & 1 & 0 & 0 & \cdots & 0 \\
0 & r & -1 & -1 & \cdots & -1 \\
-1 & -1 & 2 & 0 & \cdots & 0 \\
-1 & -1 & 0 & 2 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
-1 & -1 & 0 & 0 & \cdots & 2
\end{vmatrix},
$$

where the latter is obtained by adding every other row to the first row. Performing row operations by adding the first row to every row starting from the third row, the

determinant becomes

$$\begin{vmatrix} 1 & 1 & -1 & \cdots & -1 \\ 0 & r & -1 & \cdots & -1 \\ 0 & 0 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{vmatrix} = 2^{r-1}r.$$

Since such row operations do not change the determinant, so $\tau(K_{2,r}) = 2^{r-1}r$. □

The minimum spanning tree problem asks the following:

> Given a connected graph G, what is a spanning tree of G with the smallest possible weight, and what is this smallest weight?

More formally, we wish to find a spanning tree T of G such that $w(T) \leq w(T')$ for any spanning tree T' of G.

Intuitively, if we already have a partial minimum spanning tree of G, we can greedily choose the edge with the smallest weight from $E(G)$ such that the spanning tree is maintained after adding this edge. We introduce two famous algorithms for this purpose.

Technique 3.7.14 ▶ Kruskal's Algorithm

Let G be a graph of order n.
- Define
  - i: Current size of the partial spanning tree.
  - E_i: The set of edges contained in the partial spanning tree of size i.
- Initialise $i = 0$ and $E_0 = \varnothing$.
- Select $e \in E(G) - E_i$ such that $w(e)$ is the minimum and $[E_i \cup \{e\}]$ contains no cycle.
- Set $E_{i+1} = E_i \cup \{e\}$ and increment i.
- Repeat the above processes until $i = n - 1$, and $[E_i]$ is a minimum spanning tree of G.

Technique 3.7.15 ▶ Prim's Algorithm

Let G be a graph of order n.
- Define
  - i: Current size of the partial spanning tree.
  - S_i: The set of vertices contained in the partial spanning tree of size i.

- E_i: The set of edges contained in the partial spanning tree of size i.
- Initialise $i = 1$, $S_1 = \{u_1, v_1\}$ and $E_1 = \{u_1v_1\}$ such that $w(u_1v_1) \leq w(e)$ for all $e \in E(G)$.
- Select $uv \in E(G) - E_i$ such that $u \in S_i$, $v \notin S_i$ and $w(uv)$ is the minimum.
- Set $S_{i+1} = S_i \cup \{v\}$, $E_{i+1} = E_i \cup \{uv\}$ and increment i.
- Repeat the processes until $i = n$, and $\left[E_i\right]$ is a minimum spanning tree of G.