ENGG2440 Discrete mathematics for engineers

Ryan Chan

December 2, 2024

Abstract

This is a note for **ENGG2440 - Discrete mathematics for engineers** for self-revision purpose ONLY.
Some contents are taken from lecture notes and reference book.
Mistakes might be found. So please feel free to point out any mistakes.

Contents

1 Mathematical Induction **2**
1.1 Introduction 2
1.2 Mathematical Induction 2
1.3 Strong Mathematical Induction 3

2 Summation Techniques **5**
2.1 Summation 5
2.2 Close Form Formula 6
2.3 Perturbation Method 6
2.4 Guess-and-Verify method 8
2.5 Multiple Summation 9

3 Recurrences **12**
3.1 Homogeneous Recurrences 12
3.2 Non-homogeneous Recurrences 13

4 Asymptotics **16**
4.1 Big O 16
4.2 Big Omega 17
4.3 Theta 18
4.4 Little O 18
4.5 Little Omega 19
4.6 Properties for Asymptotic Analysis 19

5 Set Theory and Counting Principle **20**
5.1 Set Theory 20
5.2 Counting Principle 20
5.3 Permutation and Combination 23

6 Binomial Coefficients **26**
6.1 Introduction 26

7 Elements of Discrete Probability **28**
7.1 A 28

8 Introduction to Graph Theory **29**
8.1 Introduction 29
8.2 Simple Graph 31
8.3 Graph Search 32

Chapter 1

Mathematical Induction

1.1 Introduction

In mathematics, there are some basic proof techniques that we can apply, including direct proof, proof by induction, proof by contradiction, and proof by contraposition. For most of these proving methods, you won't be learning their reasons or applications, but you will still use them in some simple proving questions. In this chapter, we will mainly discuss mathematical induction.

Definition 1.1.1 (Proposition). A **proposition** is a statement that is either true or false.

Definition 1.1.2 (Predicate). A **predicate** is a proposition whose truth depends on one or more variables.

1.2 Mathematical Induction

An analogy of the principle of mathematical induction is the game of dominoes. Suppose the dominoes are lined up properly, so that when one falls, the successive one will also fall. Now by pushing the first domino, the second will fall; when the second falls, the third will fall; and so on. We can see that all dominoes will ultimately fall.

The key point is only two steps:

1. the first domino falls;
2. when a domino falls, the next domino falls.

We use the above principle of Mathematical Induction to prove.

Process:

1. Let $P(n)$ be a predicate.
2. (Base Case) Show that $P(1)$ is true.
3. (Inductive Steps) Show that for $n = 1, 2, \ldots$, if $P(n)$ is true, then $P(n+1)$ is true.

Example.

$$P(n) : 1 + 2 + \cdots + n = \frac{n(n+1)}{2}$$

1. Base Case

We need to show that $P(1)$ is true.

$$1 = \frac{(1)(1+1)}{2},$$

which is obviously true.

2. Inductive Step

For inductive hypothesis, we can assume

$$1 + 2 + \cdots + n = \frac{n(n+1)}{2}$$

Now, to show that $P(n+1)$ is true,

$$\begin{aligned} L.H.S. &= 1 + 2 + \cdots + n + (n+1) \\ &= \frac{n(n+1)}{2} + (n+1) \\ &= \frac{n(n+1) + 2(n+1)}{2} \\ &= \frac{(n+1)(n+2)}{2} \\ &= R.H.S \end{aligned},$$

which shows that $P(n+1)$ is also true.

Hence, by the principle of MI, we can conclude that $P(n)$ is true for all integers $n \geq 1$.

Exercise. Show that for any integer $n \geq 1$, $n^3 - n$ is divisible by 3.

> **Note.** In inductive step, consider putting a constant q as $3q$ is divisible by 3.

Exercise. Prove that $n^3 < 2^n$ for all integers $n \geq 10$.

> **Note.** Consider bonding the lower order terms in terms of n^3.

1.3 Strong Mathematical Induction

As the name suggests, the method of induction used in this section is "stronger". This is because assuming only that $P(n)$ is true may be too restrictive, i.e., insufficient to prove the predicate. Thus, in the inductive step, you may show that $P(1), P(2), \cdots, P(n)$ are true, and then prove that $P(n+1)$ is true.

Example. The Fibonacci sequence is a sequence of number defined via the following recursion:

$$F_n = F_{n-1} + F_{n-2},\ n \geq 2$$
$$F_0 = 0; F_1 = 1$$

Prove that

$$P(n) : F_n \leq \phi^{n-1}, \text{where } \phi = \frac{1+\sqrt{5}}{2}$$

1. Base Case

$$F_1 = 1 \leq \phi^0 = 1$$
$$F_2 = 1 \leq \phi^1 \approx 1.618$$

Thus, $P(1)$ and $P(2)$ hold true, which means $P(3)$ also holds true.

2. Inductive Step

For inductive hypothesis, we assume

$$F_k \leq \phi^{k-1} \text{ for } k = 1, 2, \ldots, n$$

Given the Fibonacci sequence

$$F_{n+1} = F_n + F_{n-1}$$

By the strong inductive hypothesis, we have

$$F_n \leq \phi^{n-1}, \quad F_{n-1} \leq \phi^{n-2}$$

From the definition of ϕ we have $\phi^2 = 1 + \phi$

Hence, we obtain

$$F_{n+1} \leq \phi^{n-1} + \phi^{n-2} = \phi^{n-2}(1 + \phi) = \phi^n$$

Chapter 2

Summation Techniques

2.1 Summation

When summing numbers with certain patterns, we can use summation notation. For example,

$$a_1 + a_2 + \cdots + a_n = \sum_{k=1}^{n} a_k$$

2.1.1 Distributive Law

Let c be a constant. Then, we can take c out of the summation:

$$\sum_{k \in \mathcal{K}} ca_k = c \sum_{k \in \mathcal{K}} a_k$$

> **Example.**
>
> $$\sum_{k=1}^{n} 2k = 2(1) + 2(2) + 2(3) + \cdots + 2(n) = 2(1 + 2 + 3 + \cdots + n) = 2 \sum_{k=1}^{n} k$$

2.1.2 Associative Law

We can split the summand as follows:

$$\sum_{k \in \mathcal{K}} (a_k + b_K) = \sum_{k \in \mathcal{K}} a_k + \sum_{k \in \mathcal{K}} b_k$$

> **Example.**
>
> $$\begin{aligned} \sum_{k=1}^{n} (k + k^2) &= (1 + 1^2) + (2 + 2^2) + \cdots + (n + n^2) \\ &= (1 + 2 + \cdots + n) + (1^2 + 2^2 + \cdots + n^2) \\ &= \sum_{k=1}^{n} k + \sum_{k=1}^{n} k^2 \end{aligned}$$

2.2 Close Form Formula

Close form formula is the formula that does not have the summation index k for a sum by simply writing it out explicitly. For example,

$$\sum_{k=1}^{n}(a_k - a_{k-1})$$

By expanding the sum, we have

$$\sum_{k=1}^{n}(a_k - a_{k-1}) = (a_1 - a_0) + (a_2 - a_1) + \cdots + (a_n - a_{n-1}) = a_n - a_0$$

By cancelling the terms, we get $a_n - a_0$, which is the close form formula for the summation $\sum_{k=1}^{n}(a_k - a_{k-1})$.

2.3 Perturbation Method

It could be difficult to derive the close form formula for some summation. Therefore, we can use the perturbation method.

For summation

$$S_n = \sum_{k=1}^{n} a_k,$$

we can split off the first term and the last term, then rewrite it as

$$a_1 + \sum_{k=2}^{n+1} a_k = S_{n+1} = \sum_{k=1}^{n} a_k + a_{n+1}$$

Example (Geometric Sum). Let x be any number. Consider the sum

$$S_n = \sum_{k=1}^{n} x^k$$

$$x + \sum_{k=2}^{n+1} x^k = S_{n+1} = \sum_{k=1}^{n} x^k + x^{n+1}$$

Observe that

$$\sum_{k=2}^{n+1} x^k = x^2 + x^3 + \cdots + x^{n+1} = x(x + x^2 + \cdots + x^n) = xS_n$$

By substitution, we have

$$x + xS_n = S_n + x^{n+1}$$

If $x \neq 1$, then we can solve for S_n to get

$$S_n = \frac{x(1 - x^n)}{1 - x}$$

This summation is also called geometric sum.

By applying the perturbation method, we can find the close form formula for some common summation. Another example is Quadratic Series.

Example (Quadratic Series). By applying perturbation method to the sum

$$S_n = \sum_{k=1}^{n} k^2,$$

we have

$$1 + \sum_{k=2}^{n+1} k^2 = S_{n+1} = \sum_{k=1}^{n} k^2 + (n+1)^2$$

Let $j = k - 1$,

$$\sum_{k=2}^{n+1} k^2 = \sum_{j=1}^{n} (j+1)^2$$

$$\begin{aligned}\sum_{j=1}^{n} (j+1)^2 &= \sum_{j=1}^{n} (j^2 + 2j + 1)\\ &= \sum_{j=1}^{n} j^2 + 2\sum_{j=1}^{n} j + \sum_{j=1}^{n} 1\\ &= S_n + 2\sum_{j=1}^{n} j + n\end{aligned}$$

Then, we have

$$1 + \sum_{k=2}^{n+1} k^2 = \sum_{k=1}^{n} k^2 + (n+1)^2$$

$$1 + S_n + 2\sum_{j=1}^{n} j + n = S_n + (n+1)^2$$

$$\sum_{j=1}^{n} j = \frac{n(n+1)}{2}$$

However, we obtain the Euler′s trick here instead. Thus, we may apply the perturbation method to another sum.

$$C_n = \sum_{k=1}^{n} k^3 \Rightarrow 1 + \sum_{k=2}^{n+1} k^3 = C_{n+1} = \sum_{k=1}^{n} k^3 + (n+1)^3$$

$$\begin{aligned}\sum_{k=2}^{n+1} k^3 &= \sum_{j=1}^{n} (j+1)^3 \quad \text{(By applying } j = k-1\text{)}\\ &= \sum_{j=1}^{n} j^3 + 3\sum_{j=1}^{n} j^2 + 3\sum_{j=1}^{n} j + \sum_{j=1}^{n} 1\\ &= C_n + 3S_n + \frac{3n(n+1)}{2} + n\end{aligned}$$

By substitution, we get

$$\begin{aligned}1 + C_n + 3S_n + \frac{3n(n+1)}{2} + n &= C_n + (n+1)^3\\ 2 + 6S_n + 3n(n+1) + 2n &= 2(n+1)^3\\ S_n &= \frac{2(n+1)^3 - 2n - 2 - 3n(n+1)}{6}\\ S_n &= \frac{(n+1)(2(n+1)^2 - 2 - 3n)}{6}\\ S_n &= \frac{n(n+1)(2n+1)}{6}\end{aligned}$$

There are some shortcut expression that might be used without finding the closed form formula on your own.

Proposition 2.3.1 (Close form formula). (You can try the perturbation method to find the close form formula by yourself.)

- Geometric series

$$\sum_{k=0}^{n} ar^k = \frac{a(r^{n+1}-1)}{r-1}, r \neq 1$$

- Euler's trick

$$\sum_{k=1}^{n} k = \frac{n(n+1)}{2} \text{ or } \sum_{k=a}^{b} k = \frac{(b-a+1)(a+b)}{2}$$

- Quadratic series

$$\sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}$$

- Cubic series

$$\sum_{k=1}^{n} k^3 = \frac{n^2(n+1)^2}{4}$$

2.4 Guess-and-Verify method

As the name suggests, we can often guess the closed-form formula. But how can we be certain it's correct? This is where mathematical induction comes in handy.

Example.

$$S_n = \sum_{k=1}^{n} k^2$$

Observe that S_n behaves like the sum of terms in a polynomial, allowing us to form an n-term polynomial. Since n^2 is the largest term in S_n, we conclude that $S_n \leq n^3$

$$S_n = a + bn + cn^2 + dn^3$$

$$\begin{aligned} 1 = S_1 &= a + b + c + d, \\ 5 = S_2 &= a + 2b + 4c + 8d, \\ 14 = S_3 &= a + 3b + 9c + 27d, \\ 30 = S_4 &= a + 4b + 16c + 64d \end{aligned}$$

Solving for above, we have $a = 0,\ b = \frac{1}{6},\ c = \frac{1}{2},\ d = \frac{1}{3}$.

$$S_n = \frac{1}{6}n + \frac{1}{2}n^2 + \frac{1}{3}n^3$$

After hypothesizing the closed-form formula, we need to use induction to verify its correctness. The steps are straightforward, and you can try proving it yourself. This process confirms that the formula is indeed the closed form for the summation.

2.5 Multiple Summation

All the summations above use only a single index. For example,

$$S_n = \sum_{k=1}^{n} k^2 = 1^2 + 2^2 + \cdots + n^2.$$

In this section, however, we introduce a summation with multiple indices. You can think of a single-index summation as summing over a 1D array. Extending this idea, a summation with two indices corresponds to summing over a 2D array. For example,

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

To sum up all the terms, we can use

$$r_j = a_{j1} + a_{j2} + \cdots + a_{jn} = \sum_{k=1}^{n} a_{jk}$$

Then, we can rewrite it as

$$S = \sum_{j=1}^{m} r_j = \sum_{j=1}^{m} \sum_{k=1}^{n} a_{jk}$$

We can also interchange the order of summation, which can be very useful for finding the closed-form formula. For the above summation, it is rather simple, we can simply do the interchange by

$$\sum_{j=1}^{m} \sum_{k=1}^{n} a_{jk} = \sum_{k=1}^{n} \sum_{j=1}^{m} a_{jk}$$

However, this does not work for all the summation.

Example. Considering

$$S = \sum_{j=1}^{n} \sum_{k=j}^{n} a_{jk}$$

To visualize that, we can again use matrix

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ & a_{22} & \cdots & a_{2n} \\ & & \ddots & \vdots \\ & & & a_{nn} \end{bmatrix}$$

Then, we have

$$\begin{aligned} S = a_{11} + & a_{12} + a_{13} + \cdots + a_{1n} \\ + & a_{22} + a_{23} + \cdots + a_{2n} \\ & \ddots \\ & + a_{nn} \end{aligned}$$

Let

$$c_k = a_{1k} + a_{2k} + \cdots + a_{nk} = \sum_{j=1}^{n} a_{jk},$$

we get

$$S = \sum_{k=1}^{n} c_k = \sum_{k=1}^{n} \sum_{j=1}^{k} a_{jk}$$

Remark. Informally, form

$$S = \sum_{j=1}^{n} \sum_{k=j}^{n} a_{jk}$$

we have $1 \leq j \leq k \leq n$, then we can simply interchange the order by

$$S = \sum_{k=1}^{n} \sum_{j=1}^{k} a_{jk}$$

Let's see another example, n-th harmonic number (H_n).

Example.

$$H_n = \sum_{k=1}^{n} \frac{1}{k}$$

Again we can visualize it by using matrix

$$\begin{bmatrix} 1 & & & & \\ 1 & 1/2 & & & \\ 1 & 1/2 & 1/3 & & \\ & & & \ddots & \\ 1 & 1/2 & 1/3 & \cdots & 1/n \end{bmatrix}$$

$$S = \sum_{j=1}^{n} H_j = \sum_{j=1}^{n} \sum_{k=1}^{j} \frac{1}{k}$$

Since $1 \leq k \leq j \leq n$

$$S = \sum_{j=1}^{n} H_j = \sum_{j=1}^{n} \sum_{k=1}^{j} \frac{1}{k} = \sum_{k=1}^{n} \sum_{j=k}^{n} \frac{1}{k}$$

Then we get

$$\begin{aligned} S &= \sum_{k=1}^{n} \sum_{j=k}^{n} \frac{1}{k} \\ &= \sum_{k=1}^{n} \frac{n-k+1}{k} \\ &= \sum_{k=1}^{n} \frac{n}{k} - \sum_{k=1}^{n} 1 + \sum_{k=1}^{n} \frac{1}{k} \\ &= (n+1)H_n - n \end{aligned}$$

Definition 2.5.1 (Floor function)**.** Given a real number x, define

$$\lfloor x \rfloor = \text{ greatest integer} \leq x$$

The symbol $\lfloor x \rfloor$ is usually read as "the floor of x".

Exercise. For any real number x, $\lfloor x \rfloor \leq x \leq \lfloor x \rfloor + 1$.

Find the closed form formula for

$$S_n = \sum_{k=1}^{n} \lfloor \sqrt{k} \rfloor$$

where, for simplicity, we assume that n is a perfect square, i.e., $n = a^2$ for some integer $a \geq 1$.

> **Remark.** Consider $\lfloor x \rfloor = \sum_{k=1}^{x} 1$, then how to do the order interchange such that we can get the following summation
>
> $$S_n = \sum_{j=1}^{a} \sum_{k=j^2}^{a^2} 1$$

Let's look at the last example.

Example. Let $n \leq 1$ be an integer and $x \neq 1$ be a real number.

$$S = \sum_{j=1}^{n} \sum_{k=1}^{j} kx^j$$

By interchanging the summation orders yields

$$S = \sum_{k=1}^{n} \sum_{j=k}^{n} kx^j$$

Notice that for any integer k satisfying $1 \leq k \leq n$ it holds that

$$\sum_{j=k}^{n} x^j = \sum_{j=1}^{n} x^j - \sum_{j=1}^{k-1} x^j = \frac{x - x^{n+1}}{1-x} - \frac{x - x^k}{1-x} = \frac{x^k - x^{n+1}}{1-x}$$

Equivalently,

$$\sum_{j=k}^{n} x^j = \frac{x^k - x^{n+1}}{1-x} = x^{k-1} \cdot \frac{x - x^{n-k+2}}{1-x} = x^{k-1} \cdot \sum_{j=1}^{n-k+1} x^j$$

Then we have

$$\begin{aligned} S &= \sum_{k=1}^{n} \sum_{j=k}^{n} kx^j \\ &= \sum_{k=1}^{n} k \sum_{j=k}^{n} x^j \\ &= \sum_{k=1}^{n} k \cdot \frac{x^k - x^{n+1}}{1-x} \\ &= \frac{1}{1-x} \cdot \sum_{k=1}^{n} kx^k - \frac{x^{n+1}}{1-x} \cdot \sum_{k=1}^{n} k \\ &= \frac{1}{1-x} \cdot \frac{x \cdot (nx^{n+1} - (n+1)x^n + 1)}{(x-1)^2} - \frac{x^{n+1}}{1-x} \cdot \frac{n(n+1)}{2} \end{aligned}$$

Chapter 3

Recurrences

Sometimes recurrences are closely related to sums. Thus, if we are going to find the closed-form formula, we can express the sum as a recurrence, then the problem will be comparatively easier.

3.1 Homogeneous Recurrences

Definition 3.1.1 (Linear homogeneous recurrence). A linear homogeneous recurrence relation of degree d with constant coefficients is a recurrence relation of the form

$$T(n) = a_1 T(n-1) + a_2 T(n-2) + \cdots + a_d T(n-d),$$

where $a_1, a_2, \cdots, a_k \in \mathbb{R}$ are given constants and $a_k \neq 0$

To solve for homogeneous recurrences with distinct root, we can use the following procedure:

1. Solve the characteristic equation to get root $r_1, \cdots, r_d$.
2. If the roots are all distinct, form, the candidate solution

$$T_0(n) = \theta_1 r_1^n + \theta_2 r_2^n + \cdots + \theta_d r_d^n$$

3. Use the initial conditions on $T(1), \ldots, T(d)$ to determine $\theta_1, \ldots, \theta_d$

Example.

$$\begin{cases} T(n) = T(n-1) + 2T(n-2) & \text{for } n \geq 2 \\ T(0) = 2, T(1) = 7 \end{cases}$$

Let $T(n) = x^n$. Then, for characteristic equation, we have

$$\begin{aligned} x^n &= x^{n-1} + 2x^{n-2} \\ x^2 &= x + 2 \quad \text{(characteristic equation)} \end{aligned}$$

The roots are $r_1 = 2, r_2 = -1$. Since they are distinct, we can form the candidate solution

$$T_0(n) = \theta_1 2^n + \theta_2 (-1)^n.$$

By using the initial conditions, we have

$$\begin{aligned} 2 &= T_0(0) = \theta_1 + \theta_2 \\ 7 &= T_0(1) = 2\theta_1 - \theta_2 \end{aligned}$$

Solving above, we have $\theta_1 = 3, \theta_= -1$. Then, we have

$$T(n) = 3 \times 2^n - (-1)^n$$

However, if the root of the characteristic equation has a multiplicity $m \geq 1$, i.e., the root is repeated for m times, then we have $T(n) = n^{m-1}x^n$. Yet the procedures are the same as solving linear homogeneous recurrence with distinct roots.

Example.

$$\begin{cases} T(n) = 2T(n-1) - T(n-2) & \text{for} n \geq 2, \\ T(0) = 0, T(1) = 1 \end{cases}$$

Characteristic equation: $x^2 = 2x - 1$.

Solving above we have $r_1 = 1$ with multiplicity $m_1 = 2$. Then we have

$$T_0(n) = \theta_1(1)^n + n\theta_2(1)^n = \theta_1 + n\theta_2$$

By using initial conditions, we have

$$\begin{aligned} 0 &= T_0(0) = \theta \\ 1 &= T_0(1) = \theta_1 + \theta_2 \end{aligned}$$

Then, it follows that the solution to the recurrences is given by

$$T(n) = n$$

Let's see another example

Example.

$$\begin{cases} T(n) = 4T(n-1) - 5T(n-2) + 2T(n-3) & \text{for} n \geq 3, \\ T(0) = 0, T(1) = 1, T(2) = 3. \end{cases}$$

Characteristic equation: $x^3 = 4x^2 - 5x + 2$

This equation has two distinct roots, $r_1 = 1, r_2 = 2$. The multiplicity of r_1 is $m_1 = 2$. Hence,

$$T_0(n) = \theta_1(1)^n + n\theta_2(1)^n + \theta_3 2^n = \theta_1 + n\theta_2 + \theta_3 2^n$$

Using initial conditions, we have

$$\begin{aligned} 0 &= T_0(0) = \theta_1 + \theta_3 \\ 1 &= T_0(1) = \theta_1 + \theta_2 + 2\theta_3 \\ 3 &= T_0(2) = \theta_1 + 2\theta_2 + 4\theta_3 \end{aligned}$$

Solving for above, we have

$$T(n) = -1 + n \times 0 + 1 \times 2^n = -1 + 2^n$$

3.2 Non-homogeneous Recurrences

A recurrence relation of the form

$$T(n) = a_1T(n-1) + a_2T(n-2) + \cdots + a_dT(n-d) + f(n)$$

is called non-homogeneous recurrences.

To solve for non-homogeneous recurrences, we can use the following procedures:

1. Solve the associated linear homogeneous recurrence.

2. Find the particular solution $T_p(n)$ to the linear non-homogeneous recurrence by examining the function class of $f(n)$.

3. Form the candidate solution $T_0(n) = T_h(n) + T_p(n)$, and use the initial conditions to find the parameters in $T_h(n)$.

In general, to solve non-homogeneous recurrence, we can consider the following particular solutions:

$f(n)$	$T_p(n)$
s	x_0
n	$x_1 n + x_0$
n^2	$x_2 n^2 + x_1 n + x_0$
s^n	$x_0 s^n$
ns^n	$(x_1 n + x_0)s^n$

Let's see some examples

Example. Consider the recurrence

$$\begin{cases} T(n) = 2T(n-1) + 1 & \text{for } n \geq 1 \\ T(1) = 1 \end{cases}$$

Let $T_0(n) = T_h(n) + T_p(n)$, where $T_h(n) = 2T(n-1)$.

For $T_h(n)$, characteristic equation: $x = 2$, then we have $T_h(n) = \theta 2^n$.

Since $f(n) = 1$, let $T_p(n) = x$,

$$\begin{aligned} x &= 2x + 1 \\ x &= -1 \end{aligned}$$

Then, we have

$$T_0(n) = T_h(n) + T_p(n) = \theta 2^n - 1$$

Using the initial conditions, we have

$$1 = T_0(1) = 2\theta - 1$$

This gives $\theta = 1$. Hence, the solution to the recurrence is given by

$$T(n) = 2^n - 1$$

Example. Consider the recurrence

$$\begin{cases} T(n) = 5T(n-1) - 6T(n-2) + 7^n & \text{for} n \geq 2 \\ T(0) = 0, T(1) = 1. \end{cases}$$

Let $T_0(n) = T_h(n) + T_p(n)$, where $T_h(n) = 5T(n-1) - 6T(n-2)$.

For $T_h(n)$, characteristic equation: $x^2 = 5x - 6$, with $r_1 = 3, r_2 = 2$.

Hence, we have $T_h(n) = \theta_1 3^n + \theta_2 2^n$, which is the homogeneous solution.

Since $f(n) = 7^n$ and $s = 7$ is not a root of the characteristic equation. Let $T_p(n) = x_0 7^n$ (particular solution),

$$\begin{aligned} x_0 7^n &= 5x_0 7^{n-1} - 6x_0 7^{n-2} + 7^n \\ x_0 &= 5x_0 7^{-1} - 6x_0 7^{-2} + 1 \\ x_0 - \frac{5}{7}x_0 + \frac{6}{49}x_0 &= 1 \\ x_0 &= \frac{49}{20} \end{aligned}$$

Then, we have

$$T_0(n) = T_h(n) + T_p(n) = \theta_1 3^n + \theta_2 2^n + \frac{49}{20}7^n$$

Using the initial conditions, we have

$$\begin{aligned}0 &= T_0(0) = \theta_1 + \theta_2 + \frac{49}{20} \\ 1 &= T_0(1) = 3\theta_1 + 2\theta_2 + \frac{343}{20}\end{aligned}$$

This gives $\theta_1 = -\dfrac{225}{20}, \theta_2 = \dfrac{176}{20}$. Hence, the solution to the recurrence is given by

$$T(n) = -\frac{225}{20}3^n + \frac{176}{20}2^n + \frac{49}{20}7^n$$

Remark. Let $f(n) = s^n$, r_1 and r_2 be the roots of the characteristic equation. Then

- If $s \neq r_1$, $s \neq r_2$, then $T_p(n) = x_0 s^n$;
- If $s = r_1$, $r_1 \neq r_2$, then $T_p(n) = x_0 n s^n$;
- If $s = r_1 = r_2$, then $T_p(n) = x_0 n^2 s^n$;

where x_0 is constant to be determined in all cases.

Example. Consider the recurrence

$$\begin{cases} T(n) = 6T(n-1) - 9T(n-2) + 3^n & \text{for} n \geq 2 \\ T(0) = 0, T(1) = \dfrac{1}{2}. \end{cases}$$

Let $T_0(n) = T_h(n) + T_p(n)$, where $T_h(n) = 6T(n-1) - 9T(n-2)$.

For $T_h(n)$, characteristic equation: $x^2 = 6x - 9$, $r_1 = 3$ with multiplicity $m_1 = 2$.

Hence, we have $T_h(n) = \theta_1 3^n + n\theta_2 3^n$, which is the homogeneous solution.

Since $f(n) = 3^n$ and $s = 3$ is also a root of the characteristic equation. Let $T_p(n) = x_0 n^2 3^n$ (particular solution),

$$\begin{aligned} x_0 n^2 3^n &= 6x_0(n-1)^2 3^{n-1} - 9x_0(n-2)^2 3^{n-2} + 3^n \\ 9x_0 n^2 &= 18x_0(n-1)^2 - 9x_0(n-2)^2 + 9 \\ 9x_0 n^2 &= 18x_0(n^2 - 2n + 1) - 9x_0(n^2 - 4n + 4) + 9 \\ 9x_0 n^2 &= 9x_0 n^2 - 18x_0 + 9 \\ x_0 &= \frac{1}{2} \end{aligned}$$

Then, we have

$$T_0(n) = T_h(n) + T_p(n) = \theta_1 3^n + n\theta_2 3^n + \frac{1}{2}n^2 3^n$$

Using the initial conditions, we have

$$\begin{aligned} 0 &= T_0(0) = \theta_1 \\ \frac{1}{2} &= T_0(1) = 3\theta_1 + 3\theta_2 + \frac{3}{2} \end{aligned}$$

This gives $\theta_1 = 0, \theta_2 = -\frac{1}{3}$. Hence, the solution to the recurrence is given by

$$T(n) = -\frac{1}{3}n3^n + \frac{1}{2}n^2 3^n$$

Chapter 4

Asymptotics

Asymptotic notation is a shorthand used to give a quick measure of the behavior of a function $f(n)$ as n grows large.

4.1 Big O

Big O is the most frequently used asymptotic notation. It is used to give an upper bound on the growth of a function, such as the running time of an algorithm.

Definition 4.1.1. We say that $f(x) = O(g(x))$ iff there exists a constant $c > 0$ and an $x_0 \geq 0$ such that

$$f(x) \leq cg(x) \quad \text{for all } x \geq x_0$$

Or we can use the following notation

$$\exists c > 0, x_0 \geq 0 \text{ such that } f(x) \leq cg(x) \forall x \geq x_0$$

Example. Let $f(x) = x^2$ and $g(x) = x^3$.

By taking $c = 1$ and $x_0 = 1$, we can simply conclude that $x^2 = O(x^3)$.

However, to prove that **there is no constant** c **such that** $x^2 \leq cx^3$ **for all** $0 \leq x \leq 1$, we need to use proof by contradiction.

For $0 \leq x \leq 1$, we have $x^2 \geq x^3$. Therefore, if such a constant $c > 0$ exists, then we must have $c \geq 1$ such that $x^2 \leq cx^3$. However, whenever $0 \leq x \leq \frac{1}{c}$, we have

$$x^3 \geq \frac{1}{c}x^2 > x^3.$$

Since we get $x^3 > x^3$, by contradiction, we know that there is no constant c such that $x^2 \leq cx^3$ for all $0 \leq x \leq 1$

We can also use differentiation to show the upper bound of a function.

Example. Let $f(x) = \ln x$ and $g(x) = x$.

Observe that $f(1) = \ln 1 = 0 < 1 = g(1)$. Moreover, $f'(x) = \frac{1}{x}$ and $g'(x) = 1$ for all $x > 0$, which implies that $g'(x) > f'(x)$ for all $x > 1$. It is the same as saying that $g(x)$ grows faster than the function $f(x)$ because the slope of the former is larger than that of the latter. It follows that

$$\ln x \leq x \text{ for all } x \geq 1.$$

Hence, by taking $c = 1$ and $x_0 = 1$ we have $\ln x = O(x)$.

Exercise. Let $f(x) = x^2$ and $g(x) = 2^x$. Show that $f(x) = O(g(x))$.

Example. Let

$$f(n) = n \cdot (n+1) \cdot (n+2) + (-1)^n, \quad g(n) = n^\alpha$$

for any integer $n \geq 1$. What is the smallest $\alpha \in \mathbb{R}$ such that $f(n) = O(g(n))$? Note that

$$\begin{aligned} |f(n)| &= |n \cdot (n+1) \cdot (n+2) + (-1)^n| \\ &= |n^3 + 3n^2 + 2n + (-1)^n| \\ &= n^3 + 3n^2 + 2n + (-1)^n \quad (\text{for any } n \geq 1) \end{aligned}$$

We claim that $\alpha = 3$ is the smallest $\alpha \in \mathbb{R}$ such that $f(n) = O(g(n))$.

Let us first prove that α satisfies $\alpha \geq 3$ and then that $\alpha = 3$.

Indeed, we must have $\alpha \geq 3$. The proof is by contradiction. If $\alpha < 3$, then

$$\lim_{n\to\infty} \frac{|f(n)|}{g(n)} = \lim_{n\to\infty} \frac{n^3 + 3n^2 + 2n + (-1)^n}{n^\alpha} = \infty,$$

which shows that $f(n) = \omega(g(n))$. However, this implies that $f(n) = O(g(n))$ is wrong. We have reached the promised contradiction.

So let us now show that for $\alpha = 3$ we have $f(n) = O(g(n))$. Indeed, choosing $c = 7$ and $n_0 = 1$, we get

$$\begin{aligned} |f(n)| &= n^3 + 3n^2 + 2n + (-1)^n \\ &\leq n^3 + 3n^2 \cdot n + 2n \cdot n^2 + n^3 \\ &= 7n^3 \\ &= cn^3 \quad (\text{for all } n \geq n_0) \end{aligned}$$

4.2 Big Omega

As we use Big O notation to express upper bound, for lower bound, we have the "Big Omega" notation.

Definition 4.2.1. We say that $f(x) = \Omega(g(x))$ iff there exists a constant $c > 0$ and an $x_0 \geq 0$ such that

$$f(x) \geq cg(x) \quad \text{for all } x \geq x_0$$

Since Big O and Big Omega are essentially "mirror image" of one another, we have

Theorem 4.2.1 (Big O vs. Big Omega).

$$f(x) = O(g(x)) \iff g(x) = \Omega(f(x)).$$

Proof. By definition of big O, $f(x) = O(g(x))$ means

$$\exists c_1 > 0, x_1 > 0 \text{ such that } f(x) \leq c_1 g(x) \quad \forall x \geq x_1,$$

which means the same as

$$\exists c_1 > 0, x_1 > 0 \text{ such that } g(x) \geq \frac{1}{c_1} f(x) \quad \forall x \geq x_1,$$

Hence, by taking $c_0 = \frac{1}{c_1} > 0$ and $x_0 = x_1 \geq 0$, we have $g(x) = \Omega(f(x))$ ■

4.3 Theta

Theta can be understood as the approximation of a function.

Definition 4.3.1. We say that $f(x) = \Theta(g(x))$ iff $f(x) = O(g(x))$ and $g(x) = O(f(x))$.

> **Remark.** It is important to note that $f(x) = \Theta(g(x))$ does not mean that $f(x) = g(x)$, it just means that
> $$\exists c_{1,2} > 0 \quad \text{such that} \quad c_1 g(x) \le f(x) \le c_2 g(x) \quad \forall x \ge x_0$$

4.4 Little O

Little O notation can be understood as the strict upper bound on the growth of a function.

Definition 4.4.1 (Little O notation). For functions $f, g : \mathbb{R} \to \mathbb{R}$, with g nonnegative, we say f is asymptotically smaller than g, in symbols,

$$f(x) = o(g(x)) \quad \text{iff} \quad \lim_{x\to\infty} \frac{f(x)}{g(x)} = 0.$$

Example. For example, let $f(x) = x, g(x) = e^x - 1$. Because

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = \lim_{x\to\infty} \frac{x}{e^x - 1} = 0.$$

Hence, we have $f(x) = o(g(x))$

Example. Let

$$f(n) = \begin{cases} 0 & \text{if } n \text{ is odd,} \\ 2 & \text{if } n \text{ is even.} \end{cases}$$

Is $f(n) = o(1)$?

Let $g(n) = 1$. Upon noting $f(n) = 1 + (-1)^n$, we see that

$$\lim_{n\to\infty} \frac{f(x)}{g(x)} = \lim_{n\to\infty} f(n).$$

However, this limit does not exist, as $f(n)$ fluctuates between 0 and 2. Thus, $f(n) \neq o(1)$.

On the other hand, if $f(n) = o(n)$? Here, let $g(n) = n$. Note that $0 \le f(n) \le 2$. It follows that

$$0 \le \frac{f(n)}{g(n)} \le \frac{2}{n}$$

for $n \ge 1$. By the sandwich theorem, we then obtain

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0.$$

Thus, $f(n) = o(n)$.

4.5 Little Omega

Little Omega notation can be understood as the strict lower bound on the growth of a function.

Definition 4.5.1 (Little O notation)**.** For functions $f, g : \mathbb{R} \to \mathbb{R}$, with g nonnegative, we say f is asymptotically smaller than g, in symbols,

$$f(x) = \omega(g(x)) \quad \text{iff} \quad \lim_{x\to\infty} \frac{g(x)}{f(x)} = 0.$$

4.6 Properties for Asymptotic Analysis

4.6.1 Rules for Asymptotic Analysis

- Transitivity

$$\text{If } f(n) = \Pi(g(n)) \text{ and } g(n) = \Pi(h(n)), \text{ then } f(n) = \Pi(h(n)), \text{ where } \Pi = O, o, \Omega, \omega, \Theta$$

- Rule of sums

$$f(n) + g(n) = \Pi(max\{f(n), g(n)\}), \text{ where } \Pi = O, \Omega, \text{ or } \Theta.$$

- Rule of products

$$\text{If } f_1(n) = \Pi(g_1(n)), f_2(n) = \Pi(g_2(n)), \text{ then } f_1(n)f_2(n) = \Pi(g_1(n)g_2(n)), \text{ where } \Pi = O, o, \Omega, \omega, \Theta.$$

- Transpose symmetry

$$f(n) = O(g(n)) \text{ iff } g(n) = \Omega(f(n)).$$

- Transpose symmetry

$$f(n) = o(g(n)) \text{ iff } g(n) = \omega(f(n)).$$

- Reflexivity

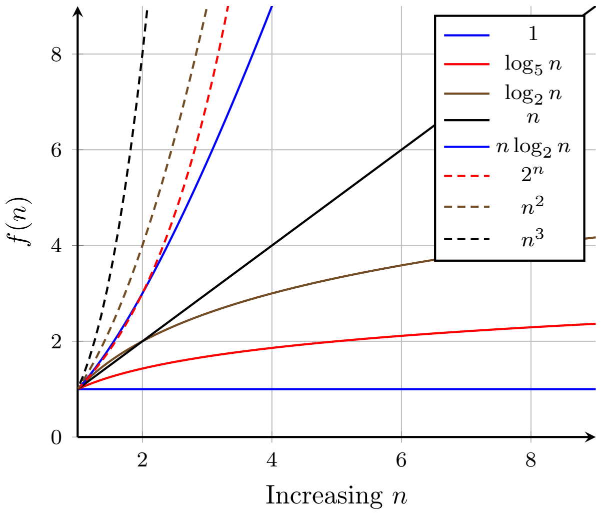
$$f(n) = \Pi(f(n)), \text{ where } \Pi = O, \Omega, \Theta.$$

- Symmetry

$$f(n) = \Theta(g(n)) \text{ iff } g(n) = \Theta(f(n)).$$

4.6.2 Graph for Functions

One can understand the growth of functions by the following graph.



Chapter 5

Set Theory and Counting Principle

5.1 Set Theory

> **Definition 5.1.1.** Set is a collection of elements, in which each element appears only once.

> **Definition 5.1.2.** Given a set S, the number of elements in S is denoted by $|S|$. The quantity $|S|$ is also referred to as the **cardinality** of S.

A set with no elements in it is called an empty set, which is denoted by $\varnothing$.

> **Definition 5.1.3.** Given two sets S and T, we say that T is a subset of S, denoted by $T \subseteq S$, if every element in T is also in S. It follows that if $T \subseteq S$, then $|T| \leq |S|$.

For example, let

$$S = \{1, 2, 3, 4, 5\}, \quad T = \{2, 4\}, \quad U = \{2, 4, 6\}$$

Then, we have $T \subseteq S$ and $T \subseteq U$.

> **Definition 5.1.4.** Let S_1 and S_2 be two given sets.
>
> The union of S_1 and S_2, denoted by $S_1 \cup S_2$, is the set containing all elements from both S_1 and S_2.
>
> The intersection of S_1 and S_2, denoted by $S_1 \cap S_2$, is the set containing all elements that are common to both S_1 and S_2.

We can also see union as the relationship "or", while intersection is the relationship "and".

> **Definition 5.1.5.** Now, let S be a set and $m \geq 1$ be an integer. A partition of S into m parts is a collection of m subsets of S, denoted by $S_1, \ldots, S_m$, with the following properties:
>
> - Exhaustion: $S = S_1 \cup S_2 \cup \cdots \cup S_m$.
> - Non-Overlapping: For $i \neq j$, we have $S_i \cap S_j = \varnothing$

5.2 Counting Principle

5.2.1 Addition Principle

Given $S = \{1, 2, 3, 4, 5\}$. Let $S_1 = \{1, 2\}, S_2 = \{3\}, S_3 = \{4, 5\}$ form a partition of S with 3 parts. Then, the cardinality of S can be determined by the cardinalities of the constituent parts, i.e.

$$|S| = |S_1| + |S_2| + \cdots + |S_m|.$$

Example (Count the number of binary strings of length n with no consecutive 1's.). Let S be the desired string, then we can apply the addition principle.

Let S_0 be the set of strings in S that start with 0; S_1 be the set of strings in S that start with 1. We have $|S| = |S_0| + |S_1|$.

We can set up a recurrence relationship. Let $|S| = T(n)$. Then, we have $|S_0| = T(n-1)$ and $|S_1| = T(n-2)$. Then we have

$$|S| = T(n) = T(n-1) + T(n-2).$$

Using the initial conditions $T(1) = 2$ (string starts with 1 or 0) and $T(2) = 3$ (if the string starts with 1, then we have only one option for the next digit; otherwise, we have two options), one can find the number of the desired binary strings.

5.2.2 Multiplication Principle

If each element in S can be generated by performing an ordered sequence of actions, say, $A_1, A_2, \dots, A_N$, and action A_i has p_i choices, where $i = 1, \dots, N$, then the cardinality of S can be computed by

$$|S| = p_1 \times p_2 \times \cdots \times p_N$$

Example. To count the number of integers between 0 and 9999 that have exactly one digit equal to 5, let S be the set of such integers. We can then partition this set to S_1, S_2, S_3, S_4, where S_1 is the set of integers in S with "5" appearing in the first position from the right, and so on. For example, $2051 \in S_2$. By the addition principle, we have

$$|S| = |S_1| + |S_2| + |S_3| + |S_4|$$

Since every integer in S_1 takes the form $xxx5$, with each "x" having 9 choices. We then have $|S_1| = 9^3$. It holds true for the other sets. Hence, we have $|S| = 4 * 9^3 = 2916$

Example. Let us count the number of odd integers between 1000 and 9999 that have all distinct digits. Let S be the set of such integers.

Method 1: Observe that each integer in S can be generated by the following ordered sequence of action:

action	choices	no. of choices
A_1 : pick the unit digit	$\{1, 3, 5, 7, 9\}$	5
A_2 : pick the tens digit	$\{0, 1, \dots, 9\}$ except the digit chosen in A_1	9
A_3 : pick the hundreds digit	$\{0, 1, \dots, 9\}$ except the digit chosen in A_1, A_2	8
A_4 : pick the thousands digit	$\{1, \dots, 9\}$ except the digit chosen in A_1, A_2, A_3	?

We cannot find the choice for A_4 directly since 0 may or may not be chosen in the previous cases. Thus, we can partition S into

S_1: set of integers in S whose tens digit is 0 $\Rightarrow |S_1| = 5 \times 1 \times 8 \times 7 = 280$,

S_1: set of integers in S whose hundreds digit is 0 $\Rightarrow |S_2| = 5 \times 8 \times 1 \times 7 = 280$,

S_3: set of integers in S with no 0 $\Rightarrow |S_3| = 5 \times 8 \times 7 \times 6 = 1680$.

Then we have $|S| = 280 + 280 + 1680 = 2240$.

Method 2: Consider a different sequence of actions:

action	choices	no. of choices
A_1 : pick the unit digit	$\{1, 3, 5, 7, 9\}$	5
A_2 : pick the thousands digit	$\{1, \dots, 9\}$ except the digit chosen in A_1	8
A_3 : pick the tens digit	$\{0, 1, \dots, 9\}$ except the digit chosen in A_1, A_2	8
A_4 : pick the hundreds digit	$\{0, 1, \dots, 9\}$ except the digit chosen in A_1, A_2, A_3	7

Then we have $|S| = 5 \times 8 \times 8 \times 7 = 2240$.

5.2.3 Subtraction Principle

Definition 5.2.1. Let S be a set and $A \subseteq S$ be a subset of S. The complement of A in S, denoted by $\overline{A}$, is the set that contains all the elements in S but not in A.

$$|S| = |A| + |\overline{A}|$$

Example. Consider computer passwords of length 6, each symbol of which is taken from 0, 1, ..., 9 and a, b, ..., z. We would like to count the number of passwords that have repeated symbols. Let A be the set of such passwords. For instance, we have $1223aq, bb333k \in A$, but $123456 \notin A$.

We can then use the subtraction principle, i.e. count the number of passwords that contains all distinct symbols. Then, we have

$$|A| = |S| - |\overline{A}| = 36^6 - P(36, 6)$$

5.2.4 Division Principle

Division principle states that if S is partitioned into k equal-sized parts, then

$$k = \frac{|S|}{\text{number of elements in each part}}.$$

Definition 5.2.2 (Ceil function). Given a real number x, define

$$\lceil x \rceil = \text{ the least integer } \geq x$$

The symbol $\lceil x \rceil$ is usually read as "the ceiling of x".

Proposition 5.2.1 (Pigeonhole Principle). Suppose that n objects are placed into k boxes. Then, at least one box has at least $\lceil \frac{n}{k} \rceil$ objects.

Example. 51 distinct numbers are chosen from the integers between 1 and 100 inclusively.

Let the 50 pairs of consecutive integers

$$\{1, 2\}, \{3, 4\}, \dots, \{99, 100\}$$

be the pigeonholes and the 51 numbers be the pigeons. Then two of the 51 numbers must be in the same pigeonholes. Therefore, there are 2 consecutive integers among the 51 chosen integers.

5.3 Permutation and Combination

5.3.1 Permutation

> **Example.** Consider a set of n elements: $S_0 = \{1, \dots, n\}$. We call S_0 the ground set. Also, let $r \geq 1$ be integer. An r-permutation of the n-elements ground set S_0 is an ordered selection of r elements from S_0. Let S be the set of all different r-permutations of the n-element ground set S_0. For instance, when $S_0 = \{1, 2, 3\}$ and $r = 2$, we have
>
> $$S = \{(1,2), (2,1), (1,3), (3,1), (2,3), (3,2)\},$$
>
> where the order of the two elements matters.

For general values of n and r. Let $P(n,r)$ denotes this number. By performing the following ordered sequence of action,

action	number of choices
A_1: pick the 1st element	n
A_2: pick the 2nd element	$n-1$
$\vdots$	$\vdots$
A_r: pick the rth element	$n-r+1$

Thus, we have

$$P(n,r) = n(n-1)\cdots(n-r+1) = \frac{n!}{(n-r)!}$$

5.3.2 Combination

> **Example.** Consider the ground set $S_0 = \{1, \dots, n\}$ of n elements. Let $r \geq 1$ be integer. An r-combination of the n-element ground set S_0 is an unordered selection of r elements from S_0. Let S be the set of all different r-combinations of the n-element ground set S_0. For instance, when $S_0 = \{1, 2, 3\}$ and $r = 2$, we have
>
> $$S = \{(1,2), (1,2), (2,3)\}$$

For general values of n and r, we can determine $\binom{n}{r}$ by relating r-combinations to r-permutations. Indeed, observe that each r-permutation of S_0 can be generated via the following ordered sequence of action:

A_1: pick r elements from S_0

A_2: order the r elements chosen from A_1 to form the desired r-permutation.

Note that A_1 has $\binom{n}{r}$ choices, while A_2 has $r!$ choices. Hence, by the multiplication principle,

$$P(n,r) = \text{number of } r\text{-permutation of } S_0 = \binom{n}{r} \times r!$$

Since

$$P(n,r) = \frac{n!}{(n-r)!},$$

it follows that

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}$$

> **Corollary 5.3.1.**
>
> $$\binom{n}{0} = 1, \binom{n}{1} = n, \binom{n}{n} = 1, \binom{n}{r} = 0 \text{ (whenever } n < r).$$

Example. Suppose that we have r presents to be distributed to k different people. How many different distributions of the presents are there? For instance, when $r = 3, k = 2$, there are four different distributions, which are given by

Person 1	Person 2
3	0
2	1
1	2
0	3

For general cases, let x_i be the number of presents received by the i-th person. Then we have

$$x_1 + x_2 + \cdots + x_k = r, x_i \geq 0.$$

The problem then becomes simply counting the number of different solutions for the above function, which is the same as counting the number of different configurations of r "1"s (we have r presents) and $k-1$ "$+$"s (to separate the presents for each of them) in $r+k-1$ placeholders. For example, when $r = 3, k = 2$, we have 4 placeholders. We then choose three of them to put an "1" and one of them to put an "$+$".

$$\underline{\quad 1 \quad} \quad \underline{\quad 1 \quad} \quad \underline{\quad + \quad} \quad \underline{\quad 1 \quad}$$

To compute the number of combinations as shown above, we can also use:

$$\binom{r+k-1}{r}$$

Example. Consider the system

$$x_1 + x_2 + x_3 + x_4 = 6, x_1 \geq 2, x_2 \geq 0, x_3 \geq 0, x_4 \geq -1.$$

We can change the variable by

$$y_1 = x_1 - 2, y_4 = x_4 + 1.$$

Then the system becomes

$$y_1 + x_2 + x_3 + y_4 = 5, y_1 \geq 2, x_2 \geq 0, x_3 \geq 0, y_4 \geq -1.$$

Then we have

$$\binom{5+4-1}{5} = \binom{8}{5}$$

Example. Consider the case when we are distributing 7 distinct presents to 3 different people, where the first and second person will get 2 presents, and the third one will get 3 presents. How many different distributions of the presents are there?

Then, for the first and second person, they will have $\binom{7}{2}$ and $\binom{5}{2}$ respectively. For the third person, they will have $\binom{3}{3}$ presents. By generalizing the arrangements, we have

$$\begin{aligned}
\text{Arrangements} &= \binom{7}{2} \times \binom{5}{2} \times \binom{3}{3} \\
&= \frac{7!}{2!(7-2)!} \times \frac{5!}{2!(5-2)!} \times \frac{3!}{3!(3-3)!} \\
&= \frac{7 \times 6}{2} \times \frac{5 \times 4}{2} \\
&= 210
\end{aligned}$$

Corollary 5.3.2 (Bookeeper Theorem)**.** Consider in general cases, in which we are distributing n distinct presents to k different people, where the i-th people will get n_i presents (here n_i is given a priori and fixed). Then we have

$$\begin{aligned}\text{Arrangements} &= \binom{n}{n_1} \times \binom{n}{n_2} \times \cdots \times \binom{n}{n_k} \\ &= \frac{n!}{n_1!(n-n_1)!} \times \frac{(n-n_1)!}{n_2!(n-n_1-n_2)!} \times \cdots \times \frac{(n-n_1-\cdots-n_{k-1})!}{n_k!(n-n-n_1-\cdots-n_{k-1})!} \\ &= \frac{n!}{n_1! \times n_2! \times \cdots \times n_k!}\end{aligned}$$

Chapter 6

Binomial Coefficients

6.1 Introduction

In this section, we introduce Binomial Coefficients.

6.1.1 Combinations

As it is introduced before,

Definition 6.1.1. An r-combination of the n-element ground set S_0 is an **unordered selection** of r elements from S_0.

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}$$

6.1.2 Permutation

Again, as it is introduced before,

Definition 6.1.2. An r-permutation of the n-element ground set S_0 is an **ordered selection** of r elements from S_0.

$$P(n,r) = \frac{n!}{(n-r)!}$$

6.1.3 Binomial Identities

Proposition 6.1.1. For any integers $m, r \geq 0$ with $0 \leq r \leq n$,

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

We have two ways to prove this proposition, namely Algebraic Proof and Combinatorial Proof.

Algebraic Proof.

$$\binom{n}{n-r} = \frac{n!}{(n-r)!(n-n+r)!} = \frac{n!}{r!(n-r)!} = \binom{n}{r}$$

■

Combinatorial Proof. Both side of the identity are supposed to be two different ways of solving a counting problem. We can define the counting problem as counting the number of different unordered selections of r elements from an n-element ground set. Then, we can define the RHS as the selecting number to be excluded. Then this identity holds. ■

6.1.4 Pascal's Identity

Theorem 6.1.1. For any integers $n,\ r \geq 0$ with $1 \leq r \leq n-1$,

$$\binom{n}{r} = \binom{n-1}{r} + \binom{n-1}{r-1} \text{ OR } \binom{n+1}{r} = \binom{n}{r} + \binom{n}{r-1}$$

Again, we can prove this theorem by two ways.

Algebraic Proof.

$$\begin{aligned} RHS &= \frac{(n-1)!}{r!(n-r-1)!} + \frac{(n-1)!}{(r-1)!(n-1-r+1)!} \\ &= \frac{(n-1)!}{r!(n-r-1)!} + \frac{(n-1)!}{(r-1)!(n-r)!} \\ &= \frac{(n-1)!}{(r-1)!(n-r-1)!}\left(\frac{1}{r} + \frac{1}{n-r}\right) \\ &= \frac{(n-1)!}{(r-1)!(n-r-1)!}\left(\frac{n}{r(n-r)}\right) \\ &= \frac{n!}{r!(n-r)!} \\ &= \binom{n}{r} \end{aligned}$$

■

Combinatorial Proof. Recall that $\binom{n}{k}$ equals the number of subsets with k elements from a set with n elements. Suppose one particular element is uniquely labeled X in a set with n elements.

To construct a subset of k elements containing X, include X and choose $k-1$ elements from the remaining $n-1$ elements in the set. There are $\binom{n-1}{k-1}$ such subsets.

To construct a subset of k elements **not** containing X, choose k elements from the remaining $n-1$ elements in the set. There are $\binom{n-1}{k}$ such subsets.

Every subset of k elements either contains X or not. The total number of subsets with k elements in a set of n elements is the sum of the number of subsets containing X and the number of subsets that do not contain X, $\binom{n-1}{k-1} + \binom{n-1}{k}$.

This equals $\binom{n}{k}$. ■

Chapter 7

Elements of Discrete Probability

7.1 A

Chapter 8

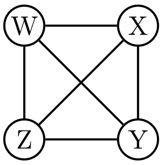
Introduction to Graph Theory

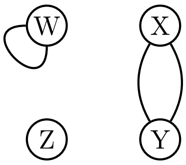
8.1 Introduction

A graph consists of a nonempty set V of vertices and a set E of edges, where each edge in E connects two (may be the same) vertices in V. We usually use $G = (V, E)$ to indicate the above relationship.

8.1.1 Simple Graph

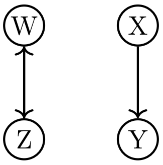
If each edge connects two different vertices, and no two edges connect the same pair of vertices, then the graph is a simple graph. For example, the graph on the left is a simple graph, and the graph on the right is not a simple graph.





8.1.2 Directed Graph

A directed graph G consists of a nonempty set V of vertices and a set E of directed edges, where each edge is associated with an ordered pair of vertices. We write $G = (V, E)$ to denote the graph. For example



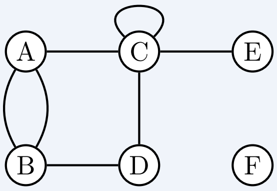
8.1.3 Undirected Graph

Let e be an edge that connects vertices u and v. We say

- e is incident with u and v
- u and v are the endpoints of e ;
- u and v are adjacent (or neighbors)
- if $u = v$, the edge e is called a loop

The degree of a vertex v, denoted by $\deg(v)$, is the number of edges incident with v, except that a loop at v contributes twice to the degree of v.

Example. Observe the following graph:

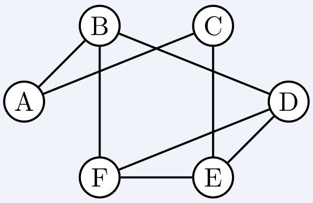


We have $\deg(A) = 3,\ \deg(B) = 3,\ \deg(C) = 5,\ \deg(D) = 2,\ \deg(E) = 1,\ \deg(F) = 0$

For a simple graph G with n vertices, if it is an undirected graph, we have a maximum of $\binom{n}{2} = \frac{n(n-1)}{2}$ edges; if it is a directed graph, then we have a maximum of $n(n-1)$ edges.

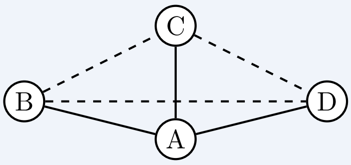
Example. Prove the proposition: among 6 people, there will be "3 mutual acquaintances" or "3 mutual strangers". Both can happen at the same time.

Consider a graph $G = (V, E)$ where V is the set of people, and E indicates acquaintance. For example,



For anyone in the graph, number of neighbors + number of non-neighbors = 5. By pigeonhole principle, we have at least $\lceil \frac{5}{2} \rceil = 3$ neighbor or non-neighbors for a person.

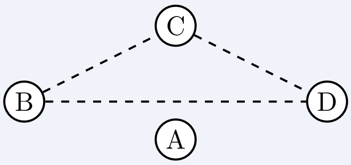
Case 1: number of neighbors of A ≥ 3. Let B, C, D be the neighbors, i.e.



If $(B, C) \in E$ or $(C, D) \in E$ or $(B, D) \in E$, then we have a triangle formed by three nodes, i.e. there are three acquaintances.

If $(B, C) \notin E$ and $(C, D) \notin E$ and $(B, D) \notin E$, then we have three mutual strangers.

Case 2: number of non-neighbors of A ≥ 3. Let B, C, D be the non-neighbors, i.e.



If $(B, C) \in E$ or $(C, D) \in E$ or $(B, D) \in E$, then we have a triangle formed by three nodes, i.e. there are three acquaintances.

Otherwise, A and at least 2 members from B, C, D will form a group of 3 mutual strangers.

8.1.4 Handshaking Theorem

Theorem 8.1.1 (Handshaking Theorem)**.** Let $G = (V, E)$ be an undirected graph with $|E|$ edges. Since each edge e contributes exactly twice to the sum on the left side (one to each endpoint).

$$\sum_{v \in V} \deg(v) = 2|E|$$

Corollary 8.1.1. An undirected graph has an even number of vertices of odd degree.

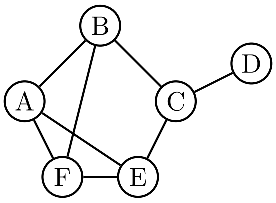
$$2|E| = \sum_{i=1}^{n} \deg(v_i) = \sum_{i:\deg(v_i)=\text{odd}} \deg(v_i) + \sum_{i:\deg(v_i)=\text{even}} \deg(v_i)$$

$\sum_{i:\deg(v_i)=\text{odd}} \deg(v_i)$ is an even number, then the number of terms summed is even. Thus, the number of odd-degree vertices is even.

8.2 Simple Graph

Definition 8.2.1 (Simple Paths)**.** A simple path in G is either a single vertex or an ordered list of distinct vertices $v_0 - v_1 - \cdots - v_k$.

Consider the following graph



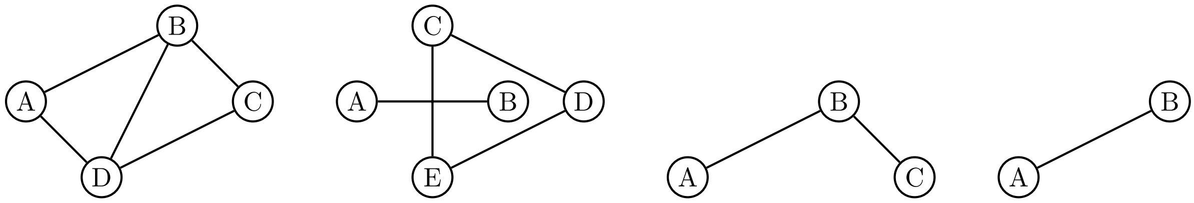
We then have simple path from C to F: $C - E - F$. However, $C - B - C - E - F$ is not a simple path.

Definition 8.2.2 (Cycles)**.** A cycle in G is a path $v_0 - v_1 - \cdots - v_k$ such that $v_0 = v_k$ and $k \geq 3$.

Again, by observing the graph above, we can see that one example for cycle is $C - E - A - B - C$. However, $C - D - C$ is not a cycle. The length of a path/cycle is the number of edges in the path.

8.2.1 Properties of Graphs

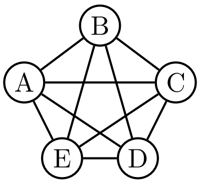
- A graph is connected if every pair of vertices has a path between them. (fig.1 is connected, fig.2 is not connected)
- A graph is acyclic if it does not contain cycle. (fig.3, fig.4)
- A connected, acyclic graph is called a tree. (fig.3)
- A leaf of a tree is a vertex of degree1. (both vertices in fig.4 are leaves)



(a) fig.1 (b) fig.2 (c) fig.3 (d) fig.4

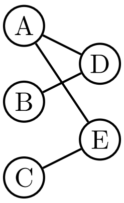
8.2.2 Special Simple Graphs

A complete graph on n vertices, denoted by K_n, is a simple graph that contains one edge between each pair of distinct vertices. For example, for graph K_5, we have:



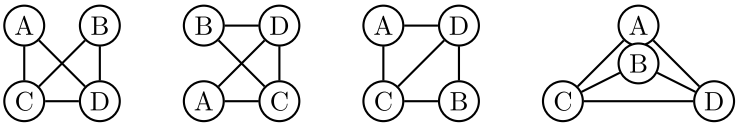
An n-cube, denoted by Q_n, is a graph that consists of 2^n vertices, each representing a distinct n-bit string. An edge exists between two vertices is the corresponding strings differ in exactly one bit position.

A bipartite graph is a graph such that the vertices can be partitioned into two sets V and W, so that each edge has exactly one endpoint from V, and one endpoint from W. For example, in the following graph, $A, B, C \in V_1,\ D, E \in V_2$ such that $V = V_1 \cup V_2,\ V_1 \cap V_2 = \varnothing$. One can also assign one of two different colors to each vertex, so that no adjacent vertices are assigned the same color.



8.2.3 Graph Isomorphism

A graph $G = (V_1, E_1)$ and $H = (V_2, E_2)$ are isomorphic if we can set up a bijection $f : V_1 \to V_2$ such that x and y are adjacent in G. For example, the following graphs are isomorphic to each other.

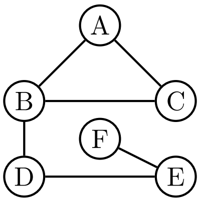


By observation, one can see that if two graphs are isomorphic, then the adjacent vertices in the original graph will still be adjacent, and the degree of such vertices also remains unchanged.

8.3 Graph Search

8.3.1 Eulerian Paths and Circuits

In graph theory, an Eulerian path in a graph is a path that contains each edge exactly once (allowing for revisiting vertices). If such a path is also a circuit, it is called an Eulerian circuit. For example, the following graph contains an Eulerian path:



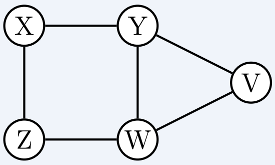
Remark. A connected graph G has an Eulerian circuit if and only if each vertex of G has even degree.

8.3.2 Graph Representation

We can use the adjacency matrix or adjacency list to represent an undirected graph.

Example. Below shows two types of graph representation:

Undirected Graph



Adjacency Matrix

$$\begin{array}{c|ccccc} & X & Y & Z & W & V \\ \hline X & 0 & 1 & 1 & 0 & 0 \\ Y & 1 & 0 & 0 & 1 & 1 \\ Z & 1 & 0 & 0 & 1 & 0 \\ W & 0 & 1 & 1 & 0 & 1 \\ V & 0 & 1 & 0 & 1 & 0 \end{array}$$

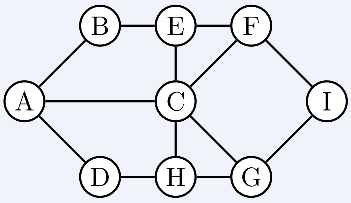
Adjacency List

[0] X → Y → Z
[1] Y → X → W → V
[2] Z → X → W
[3] W → Y → Z → V
[4] V → Y → W

8.3.3 Breadth-First Search (BFS)

For Breadth-First Search, we have graph $G = (V, E)$ as input, and source vertex $s \in V$. Then, the distance ($d[u]$) from s to u, for all $u \in V$, and $\pi[u]$, which is the predecessor of u are the outputs. One can take the idea of Breadth-First Search as a wave spread out from one vertex.

Example. Consider the following graph.



We have the following traversal: A → B → C → D → E → F → H → I → G.

We can also use the following adjacency list to show the same result

Node	Queue
A	(1, B), (1, C), (1, D)
B	(1, C), (1, D), (1, E)
C	(1, D), (1, E), (1, F), (1, G), (1, H)
D	(1, E), (1, F), (1, G), (1, H)
E	(1, F), (1, G), (1, H)
F	(1, G), (1, H), (1, I)
G	(1, H), (1, I)
H	(1, I)
I	-

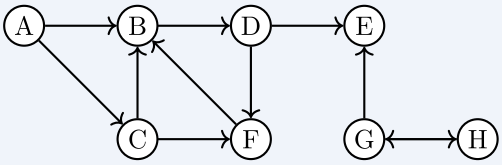
Then, for predecessor and distance, we have:

u	A	B	C	D	E	F	G	H	I
$\pi[u]$		A	A	A	B	C	C	C	F
$d[u]$	0	1	1	1	2	2	2	2	3

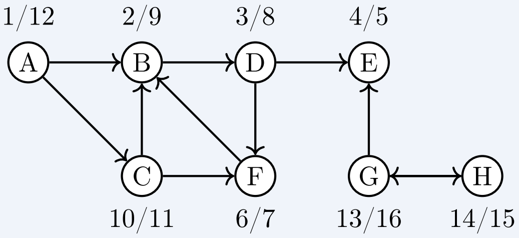
8.3.4 Depth-First Search (DFS)

For Depth-First Search, we have graph $G = (V, E)$ as input, and source vertex $s \in V$. Then, the discovery time ($d[v]$) from s to v, for all $v \in V$, and $f[v]$, which is the finishing time are the outputs. Depth-First Search is just like when we discover a vertex, we explore from it as far as we can.

Example. Consider the following graph.



By using Depth-First Search, we have the following traversal:



We use the notation "$d[v]/f[v]$" to show the discovery and finishing time. For example, the discovery time of node A is 1, finishing time is 12, then we use "1/12" to denote this timestamp. For discovery time and finishing time, we also have:

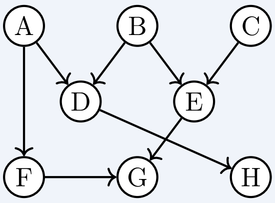
v	A	B	C	D	E	F	G	H
$d[v]$	1	2	10	3	4	6	13	14
$f[v]$	12	9	11	8	5	7	16	15

A graph is said to be strongly connected if every vertex is reachable from every other vertex. The strongly connected components of an arbitrary directed graph form a partition into subgraphs that are themselves strongly connected.

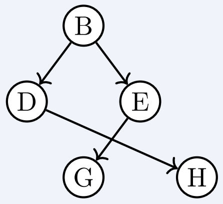
A vertex v is an articulation point (also called articulation vertex) if removing v increases the number of connected components.

A topological sort or topological ordering of a directed graph is a linear ordering of its vertices such that for every directed edge uv from vertex u to vertex v, u comes before v in the ordering. Topological sorting can only be used on directed acyclic graphs. If a graph contains cycles, it cannot be topologically sorted.

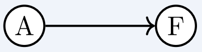
Example (Topological Sort). Two methods for solving topological sort will be introduced. Consider the following graph.



We can use DFS to find the topological order. We first select an unvisited node; for example, we choose node **B** in this case.



We have: B → D → E → H → G Then, we choose another node, in this case we choose node **A**.



We have: A → F → B → D → E → H → G.

Finally, we have C → A → F → B → D → E → H → G.

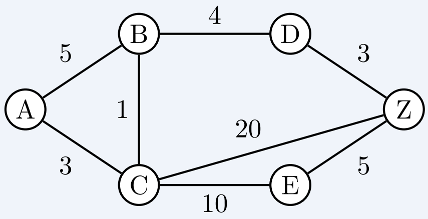
This is one of the possible topological sorts.

8.3.5 Shortest Path Problem

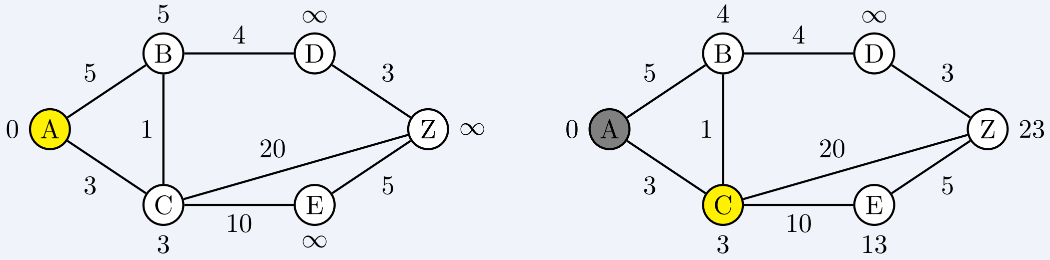
Consider a weighted graph, which has weighted edges. We define such a graph by $G = (V, E, w)$, where the newly introduced parameter w is the weight of an edge. Then how do we find the shortest path on a large graph from one point to another? We can use Dijkstra's Algorithm.

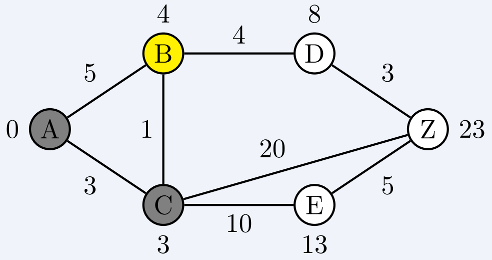
We initialize the algorithm by (A) maintain a table of cost $c(v)$, where starting vertex has cost $c(v_0) = 0$, and other vertices have cost $c(v_i) = \infty, i \neq 0$; (B) maintain a set of visited vertices $S = \varnothing$. After choosing the unvisited vertex with minimum cost, we update the cost of its adjacent vertices. After updating, we repeat this process. When every vertex is visited, the algorithm will then end. These operations are called relaxation.

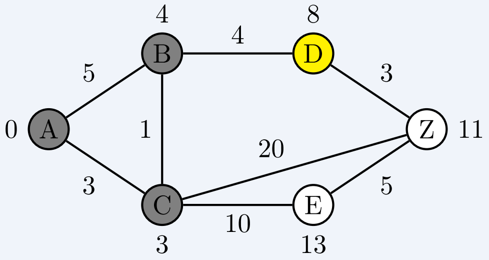
Example (Shortest Path Problem)**.** Consider the following graph, find the shortest path from vertex A to vertex Z.

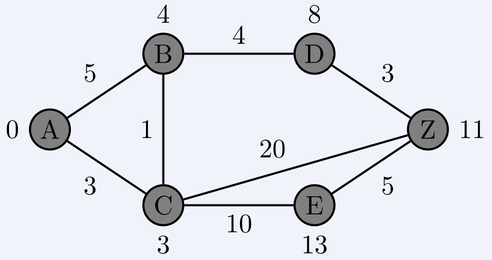


To solve this problem, we can use the method mentioned above:









We can also use tables to represent the algorithm:

	$c(\cdot)$	Path
A	0	A
B	∞	
C	∞	
D	∞	
E	∞	
Z	∞	

$S = \varnothing$

	$c(\cdot)$	Path
A	0	A
B	5	A, B
C	3	A, C
D	∞	
E	∞	
Z	∞	

$S = A$

	$c(\cdot)$	Path
A	0	A
B	4	A, C, B
C	3	A, C
D	∞	
E	13	A, C, E
Z	23	A, C, Z

$S = A, C$

	$c(\cdot)$	Path
A	0	A
B	4	A, C, B
C	3	A, C
D	8	A, C, B, D
E	13	A, C, E
Z	23	A, C, Z

$S = A, C$

	$c(\cdot)$	Path
A	0	A
B	4	A, C, B
C	3	A, C
D	8	A, C, B, D
E	13	A, C, E
Z	11	A, C, B, D, Z

$S = A, C, B, D$

	$c(\cdot)$	Path
A	0	A
B	4	A, C, B
C	3	A, C
D	8	A, C, B, D
E	13	A, C, E
Z	11	A, C, B, D, Z

$S = A, C, B, D, Z$

Therefore, the shortest path from A to Z is $A \to C \to B \to D \to Z$

8.3.6 Minimum Spanning Trees (MST)

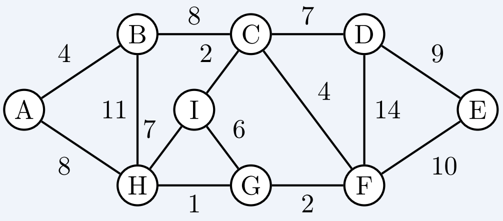
Given an undirected graph $G = (V, E)$ with weights on the edges, a minimum spanning tree (MST) of G is an acyclic subset $T \subseteq E$ that connects all nodes in V and whose total weight $w(T) = \sum_{(u,v)\in T} w(u, v)$ is minimum.

A minimum spanning tree has precisely $n - 1$ edges, where n is the number of vertices in the graph.

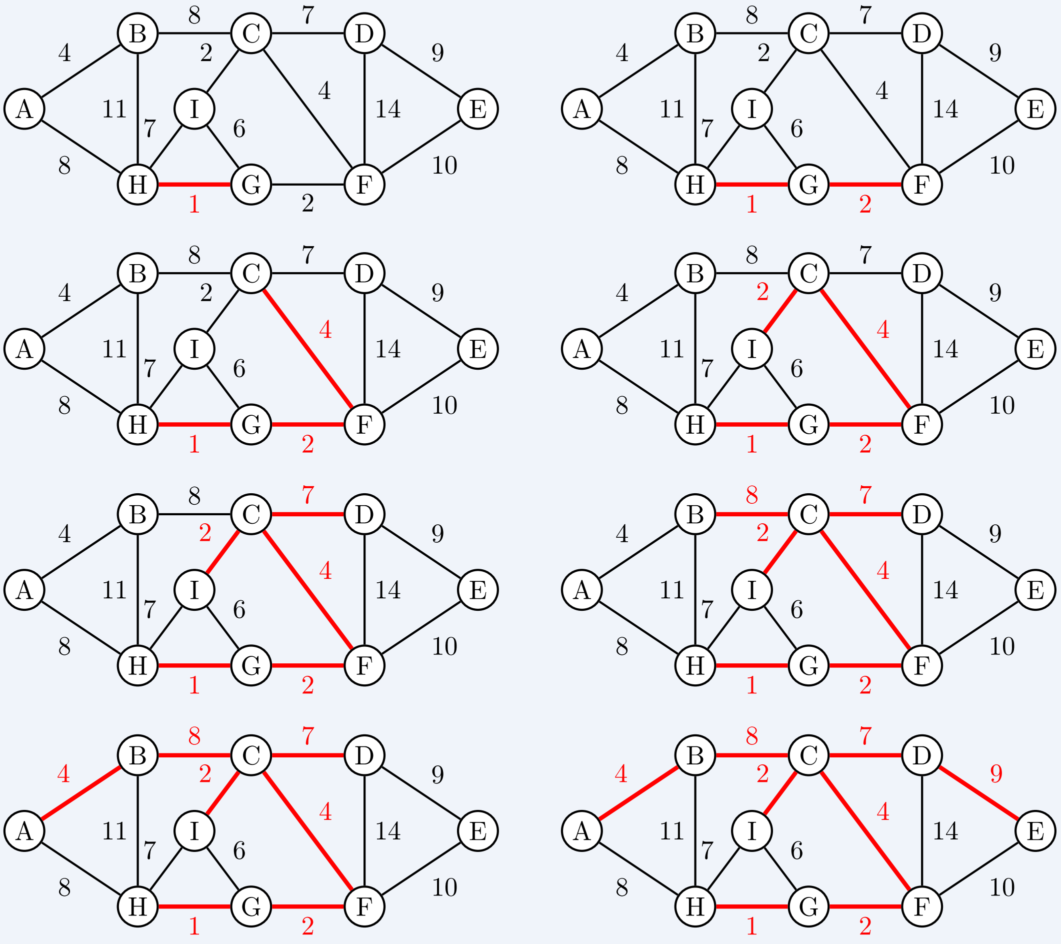
We have two algorithms for finding the MST. Prim's algorithm starts from finding the minimum edges among all edges, then finds the minimum edges that are connected to the vertices that are adjacent in the previous edges selected.

Kruskal's algorithm again starts from finding the minimum edges among all edges. However, we will keep finding the minimum edges until all vertices are being visited.

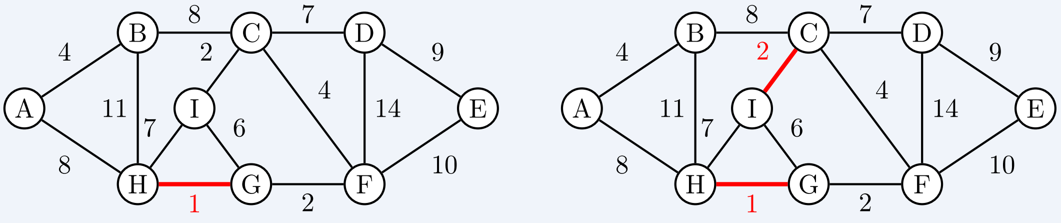
Example. Find the minimum spanning tree for the following graph.

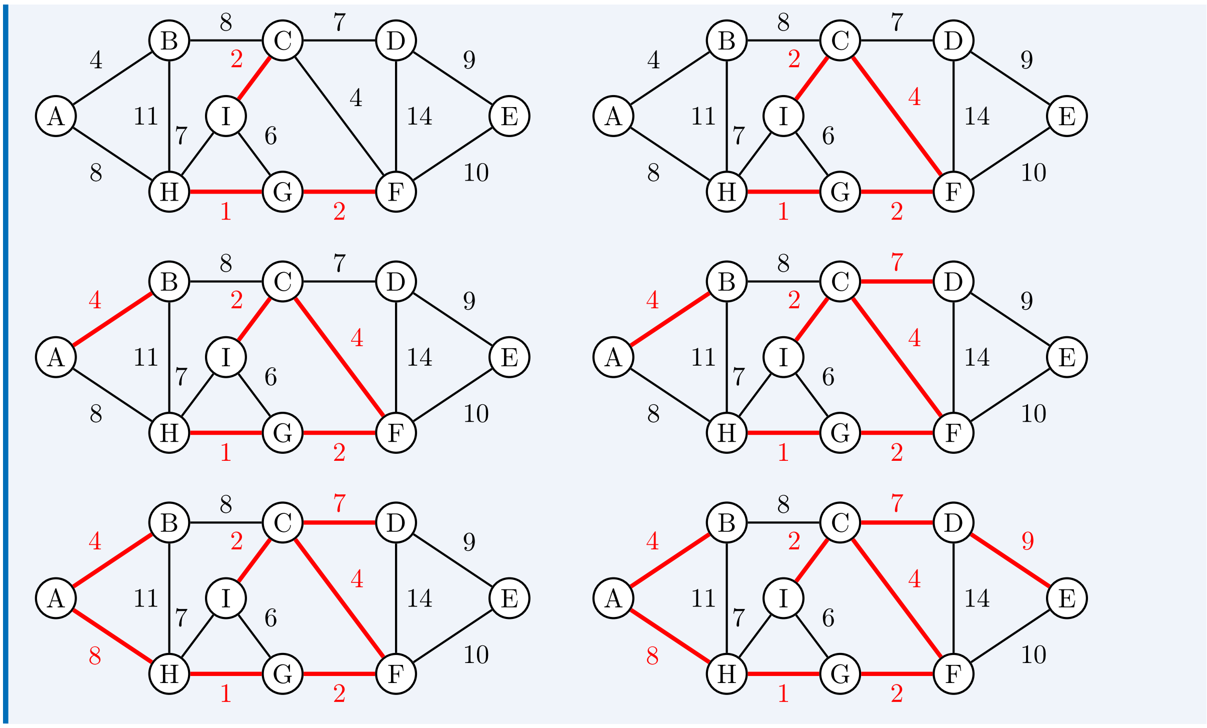


Prim's Algorithm



Kruskal's Algorithm




A
B
C
D
E
F
G
H
I
4
8
7
9
11
2
4
14
7
6
8
10
1
2