Discrete Mathematics II

MACM 201 Course Notes

Department. of Mathematics, SFU

Fall 2021

SELF PUBLISHED

http://www.math.sfu.ca

First printing, August 2020

Contents

1. Introduction to the Course

Welcome to MACM201. In this section we outline the content of the course, and end with a counting exercise.

1.1 Course Overview

Textbook: *Discrete and Combinatorial Mathematics: An Applied Introduction*, 5th edition by R. P. Grimaldi. Published by Pearson.

Prerequisites: MACM 101, with a grade of C- or better.
Topics you should be familiar with:

- Counting (Section 1.1)
- Logic (Chapter 2)
- Set Theory (Sections 3.1, 3.2 and 3.3)
- Induction and Number Theory (Chapter 4)
- Relations and Functions (Chapter 5)

Topics covered in MACM 201:

Counting:

- Principles (Section 1.2, 1.3, 1.4)
- Discrete Probability (Sections 3.4, 3.5, 3.6, 3.7)
- Generating Functions (Sections 9.1, 9.2, 9.5) – uses Functions extensively
- Recurrence Relations (Sections 10.1, 10.2, 10.3, 10.4, 10.6) – uses Induction

Graph Theory:

- Graph topics (Sections 11.1, 11.2, 11.3, 11.4, 11.5)
- Trees (Sections 12.1, 12.2, 12.5)
- Optimization (Sections 13.2)

1.2 Warm-up Counting Exercise

In the chair lift line-up at Mount Seymour they ask that visitors get into groups of 4 prior to loading onto the chair (which seats four people across). A natural questions arises, *how many ways are there to make a group of* 4?

Before answering this questions we should first think about which configurations we would consider different, and which ones we consider the same. Imagine someone saying the group compositions as they scan across the chair from left to right. They may say

'single, double, single', or 'double, single, single', or 'quadruple'.

The phrase uttered captures the configuration completely, so we are really considering the number of possible uttered phrases.
With this clarification go ahead and work out an answer to the question:

How many ways are there to make a group of 4?

What if the chair could hold 6 people, how many ways are there to make 6's?

As mathematicians we like to generalize results. Why? Well, suppose the way we solved the problem of making 4's by listing them. This approach gets more difficult as the values get larger, and just won't work at all in the case of 100 seats. This forces us to construct more elegant solutions, and in the process get a better understanding of the original problem and it's relationship to other combinatorial objects we can count.

So even though there does not exist a chair lift that seats 100 people across it is nevertheless interesting to ask the question: how many ways are there to make 100's?

Part One: Objects, Counting, and Probability

2. Basic Objects and Counting Techniques

In this chapter we lay down the foundations for this course. Our main objective is to learn how to count like a professional (advanced *enumeration techniques*), to do that we must (i) introduce a collection of objects that we will be counting (sequences, strings, permutations, graphs, trees, etc.), and (ii) introduce some basic counting techniques (rule of sum, rule of product, permutations, combinations). Graph Theory will be covered more thoroughly at the end of the course, it is introduced here just as an object we can practice our counting techniques on.

2.1 Fundamental Combinatorial Objects

2.1.1 Lecture 1

Lecture outline

Discrete mathematics differs from continuous mathematics (calculus) in that it is the mathematics of objects composed of a **finite** set of elements arranged into a specific **structure**.

Combinatorial objects are defined by:

- What are the elements (atoms) that compose them?
- How are they structured?

We will study four main types of combinatorial objects:

1. sets and subsets
2. strings and permutation
3. graphs
4. trees

The notion of an *atom* naturally leads to a notion of an objects **size**, defined as the (integer) **number of atoms** the object contains, e.g., number of letters in a string.

Example 2.1 Sets and Subsets

Strings

Definition 2.1.1 — alphabet and string. An **alphabet** Σ is a set of n elements called **letters**. A **string** S of size m is an ordered sequence of m letters from Σ.

Example 2.2
$\Sigma = \{0,1\}$

$\Sigma = \{A,C,G,T\}$

Exercise 2.1 How many DNA sequences are there of length n? Note: It's not 4^n because the DNA sequence $ACCT$ is the same as the DNA sequence $TCCA$. ■

Example 2.3 Find all strings of length 6 over $\{0,1\}$ that do not have 10 as a substring.

Permutations

Definition 2.1.2 A **permutation** P over an alphabet Σ is a string over Σ where every letter of Σ occurs exactly once.

Example 2.4 For $\Sigma = \{1,2,3\}$ find all permutations over Σ.

Graphs

Definition 2.1.3 — graph. A (simple) **graph** is a pair (V,E) where V is a set of **vertices** and E is a set of unordered pairs of vertices called **edges**. If $e = \{i,j\} \in E$ we say vertices i and j are **adjacent**. The **degree** of a vertex i in V is the number of vertices in V adjacent to vertex i.

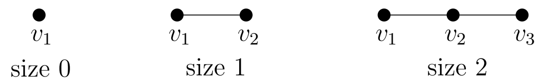
Example 2.5 $V = \{1,2,3,4,5,6\}$, $\quad E = \{\{1,2\},\{1,5\},\{2,3\},\{2,5\},\{3,4\},\{4,5\},\{4,6\}\}$

Example 2.6 How many edges can a graph with n vertices have?

Definition 2.1.4 — complete graph. A graph $G = (V,E)$ is **complete** if $|V| \geq 1$ and for all $i,j \in V$, with $i \neq j$, the edge $\{i,j\} \in E$. The **complete graph** with n vertices is denoted by K_n.

Definition 2.1.5 — path graph. A graph $G = (V,E)$ is a **path** if $|V| \geq 1$ and V may be ordered $v_1, v_2, \ldots, v_n$ so that $E = \Big\{\{v_1,v_2\},\{v_2,v_3\},\ldots,\{v_{n-1},v_n\}\Big\}$. The path graph with n vertices is denoted P_n.

Path Examples:



Definition 2.1.6 — cycle graph. A graph $G = (V,E)$ is a **cycle** if $|V| \geq 3$ and V may be ordered $v_1, v_2, \ldots, v_n$ so that $E = \Big\{\{v_1,v_2\},\{v_2,v_3\},\ldots,\{v_{n-1},v_n\},\{v_n,v_1\}\Big\}$. The cycle graph with n vertices is denoted C_n.

Example 2.7 A path of size 4 in K_5:

Example 2.8 A cycle of size 5 in K_5:

Definition 2.1.7 — connected graph. A graph $G = (V,E)$ is **connected** if there is a path in G from vertex $i \in V$ to vertex j for all $i \neq j$.

Definition 2.1.8 — tree. A graph $G = (V,E)$ is a **tree** if it is connected and has no cycles.

Example 2.9 Draw all (unlabelled) trees with 4 vertices.

Example 2.10 Draw all (unlabelled) trees with 5 vertices.

Exercise 2.2 How many labelled trees are there with 5 vertices. ■

Exercise 2.3 If G is a tree with $n > 0$ vertices, how many edges must G have? ■

2.1.2 Additional Comments

We will mostly study the families of objects introduced in this section. We might see other kinds (compositions, partitions, lattice paths, ...), but they will always be related to strings, graphs or trees. As for all combinatorial objects, it is important to remember they are formed of **atoms**, organized with some specific **structure**.

We will study various kinds of questions:

- Answering **counting** problems, mostly, for a given family $\mathscr{C}$ of combinatorial objects and a given integer n, how many objects of $\mathscr{C}$ of size n are-there? This is some kind of combinatorial calculus.
- Proving **structural properties** of a given family of combinatorial objects. We will write proofs (in general short and simple).
- Studying **algorithms** that operates on combinatorial objects (mostly trees and graphs).

It is important to understand right now that the purely "calculatory" part of MACM201 (mostly answering counting questions) is not the only important part. An important goal of this course is that you become familiar and comfortable with the combinatorial objects themselves, their properties, how to analyze them, generate them, ... This will benefit to you in several ways:

- You will have to work with these objects in later courses in computing and engineering.
- If you end up working as a scientist, computer scientist or engineer, you will very likely have to deal with discrete models that will be based on combinatorial objects.

2.2 Basic Counting Principles

2.2.1 Lecture 2 (Grimaldi 1.1, 1.2)

Basic Counting Principles

Definition 2.2.1 — Rule of Sum. If there are m ways to perform to perform task X and n ways to perform task Y, there are $m+n$ ways to perform **either** X or Y.

Definition 2.2.2 — Rule of Product. If there are m ways to perform to perform task X and n ways to perform task Y, there are mn ways to perform **both** X or Y.

Example 2.11

Example 2.12 There were 10 people at a party, and everyone hugs each other. How many hugs are there?

Basic Counting Principles: Strings

Theorem 2.2.1 — strings. If Σ is an alphabet with k letters, the number of strings of length n over Σ is k^n.

Proof.

Basic Counting Principles: Permutations

An important application of the product rule is to count the number of permutations of n objects.

Theorem 2.2.2 The number of permutations of a set of n objects is $n!$.

Proof.

Basic Counting Principles: Permutations with Repetition

Now we consider what happens when we allow for repeated indistinguishable objects?

Definition 2.2.3 — permutations with repetition. Suppose there are k_1 objects of type A, k_2 objects of type B, $\ldots$, and k_r of type R, and let $n = k_1 + k_2 + \ldots k_r$ be the total number of objects. The number of distinct permutations is denoted by $\binom{n}{k_1, k_2, \ldots, k_r}$.

Note: If $k_1 = k_2 = \cdots = k_r = 1$ then this is just the number of permutations of a set of n objects.

Example 2.13 Consider the letters M, E, E, N, N. How many permutations are there?

Theorem 2.2.3 — permutations with repetition.

$$\binom{n}{k_1, k_2, \ldots, k_r} = \frac{n!}{k_1!k_2!\ldots k_r!}$$

Proof.

Exercise 2.4 How many binary strings of length 20 are there with exactly 13 1's?

■

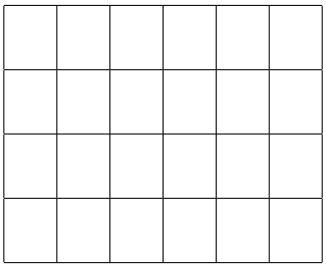
Theorem 2.2.4 — Subsets and Combinations. If S is a set of size n, the number of subsets of size k is

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

Proof.

Basic Counting Principles: Lattice Paths

Example 2.14 Many models in theoretical physics involve *lattice paths*: these are paths in the square lattice with a prescribed set of allowed steps. Here we consider paths with North and East steps.



How many lattice paths there are from the point of coordinates $(0,0)$ to the point of coordinates $(6,4)$ if we are restricted to North steps and East steps only?

2.2.2 Additional Notes: Counting Words

Here we provide an alternate proof to Theorem 2.2.3. However, before diving into full generality we will prove the following special case. (Note: if you understand this special case, all that is required for the general is a few more symbols.)

Proposition: The number of strings over the alphabet $\Sigma = \{A, B, C\}$ with exactly 4 copies of A, 3 copies of B, and 2 copies of C (so $4+3+2=9$ letters in total) is equal to $\binom{9}{4,3,2}$.

Proof. Let W denote the set of all strings over Σ with the desired properties. Define

$$\Sigma^* = \{A^1, A^2, A^3, A^4, B^1, B^2, B^3, C^1, C^2\}$$

Note that the superscripts above do not indicate powers, we are just treating Σ^* as a set of symbols; we could use subscripts instead but that would cause conflict with the notation in the more general theorem coming next. Define W^* to be the set of all permutations of Σ^*. Now we will define a function f from W^* to W by the rule that f removes the superscript from each letter (ex. $f(A^1C^2B^1A^3A^2B^3C^1A^4B^2) = ACBAABCAB$).

The first step in our analysis is to determine for an arbitrary string $S \in W$ how many strings in W^* map to S under the function f. In other words, we want to determine, for each string of W how many ways we can add superscripts to get a word in W^*. In our particular example, each word in W has 4 copies of the letter A, and we have to add the superscripts $1,2,3,4$ to them. This can be done in exactly 4! ways (the possibilities here are just the 4! permutations of $\{1,2,3,4\}$). We have to add the superscripts $1,2,3$ to the three copies of B and we can do this in 3! ways, and there are 2! ways to add superscripts to the two C's. So, in total there are exactly 4!3!2! ways to add superscripts to a given word of W to get a word in W^*.

The function $f : W^* \to W$ is a many-to-one function, and for each word $w \in W$ there are 4!3!2! strings in $f^{-1}(W) \subset W^*$. Hence $|W^*| = 4!3!2! \cdot |W|$. Since $|W^*|$ is the number of permutation of Σ^* then $|W^*| = 9!$, therefore $|W| = \frac{9!}{4!3!2!} = \binom{9}{4,3,2}$. ■

Now we prove the main result, which is just a restatement of Theorem 2.2.3.

Theorem 2.2.5 Let $\Sigma = \{A_1, \dots, A_r\}$ be an alphabet and let $k_1, \dots, k_r$ be nonnegative integers that sum to n. The number of words over Σ with exactly k_i copies of the letter A_i for every $1 \le i \le r$ is equal to

$$\binom{n}{k_1, k_2, \dots, k_r} = \frac{n!}{k_1!k_2!\dots k_r!}$$

Proof. Let W be the set of words satisfying the above restrictions. Define the alphabet

$$\Sigma^* = \{A_1^1, \dots, A_1^{k_1}, A_2^1, \dots A_2^{k_2}, \dots, A_r^1, \dots A_r^{k_r}\}.$$

and let W^* be the set of permutations over Σ^*. Define a function f from W^* to W by the rule that f replaces each letter of the form A_i^j with the letter A_i (so f drops the superscripts on the letters). For every string $S \in W$ there are exactly $k_1!k_2!\cdots k_r!$ strings in W^* that map to S, since for each letter A_i we can add back the superscripts $1, 2, \dots, k_i$ in exactly $k_i!$ ways. Now, the function.f gives a one-to-many correspondence between W and W^*. The size of W^* is the number of permutations of Σ^* and $|\Sigma^*|. = n$, so we have $|W^*| = n!$. It follows from this and our correspondence that

$$|W| = \binom{n}{k_1, k_2, \dots, k_r} = \frac{n!}{k_1!k_2!\dots k_r!}.$$

■

2.3 Combinations and The Binomial Coefficient

2.3.1 Lecture 3 (Grimaldi 1.3)

The numbers $\binom{n}{k}$

Reminder 2.3.1 The quantity $\binom{n}{k}$ is the number of ways of choosing a set of size k from a set of size n. We also saw that it is the number of binary strings of length n with exactly k 1's so

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

Theorem 2.3.1

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

Proof.

Theorem 2.3.2 For every nonnegative integer n

$$\sum_{k=0}^{n} \binom{n}{k} = \binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n} = 2^n.$$

Proof.

Expanding $(x+y)^n$

The Binomial theorem

Theorem 2.3.3 — Binomial Theorem. If x and y are two variables and n a positive integer, then,

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k} = \binom{n}{0} x^0 y^n + \binom{n}{1} x^1 y^{n-1} + \binom{n}{2} x^2 y^{n-2} + \cdots + \binom{n}{n} x^n y^0.$$

Because of this theorem the numbers $\binom{n}{k}$ are called **binomial coefficients** .

We now have three equivalent ways to think of $\binom{n}{k}$:

Using the Binomial Theorem

Exercise 2.5 Find the coefficient of x^5y^{95} in $(3x-y)^{100}$.

Multinomial Theorem

Theorem 2.3.4 — Multinomial theorem. If $x_1, x_2, \ldots, x_m$ are variables and n a positive integer, then,

$$(x_1+x_2+\cdots+x_m)^n = \sum_{k_1+k_2+\cdots+k_m=n} \binom{n}{k_1,k_2,\ldots,k_m} x_1^{k_1}x_2^{k_2}\cdots x_m^{k_m}$$

The proof is a straightforward generalization of that for the Binomial theorem.
Proof:

Exercise 2.6 What is the coefficient of xy^2z^2 in $(w+x+y+z)^5$?

2.4 Combinations with Repetition

2.4.1 Lecture 4 (Grimaldi 1.4)

Example 2.15 How many combinations of size 3 are there from $S = \{a, b, c\}$ if repetitions are allowed?

Theorem 2.4.1 — Combinations with Repetitions. Let S be a set with n elements. The number of ways to select k objects from S, with repetition allowed, is

$$\binom{n+k-1}{k} = \frac{(n+k-1)!}{k!(n-1)!}.$$

Proof with binary strings.

Example 2.16 How many integer solutions are there to

$$x_1 + x_2 + x_3 + x_4 + x_5 = 10, \quad x_i \geq 0 \ ?$$

Example 2.17 How many integer solutions are there to

$$x_1 + x_2 \leq 7, \quad \text{with } x_1 \geq 0 \text{ and } x_2 \geq 0 \ ?$$

Example 2.18 How many ways are there to distribute 5 apples, 4 oranges, and 3 pears to three people?

Example 2.19 Consider the following code segments.
What is the value of counter after the loops have executed?

Code segment 1: C

```c
counter = 0;
for( i=1; i<=20; i++ )
    for( j=1; j<=20; j++ )
        for( k=1; k<=20; k++ )
            counter = counter +1;
```

Code segment 1: Python

```python
counter = 0
for i in range(1,21):
    for j in range(1,21):
        for k in range(1,21):
            counter = counter + 1
```

Code segment 2: C

```c
counter = 0;
for( i=1; i<=20; i++ )
    for( j=i; j<=20; j++ )
        for( k=j; k<=20; k++ )
            counter = counter +1;
```

Code segment 2: Python

```python
counter = 0
for i in range(1,21):
    for j in range(i,21):
        for k in range(j,21):
            counter = counter + 1
```

Exercise 2.7 A box contains 10 red balls, 10 green balls and 10 blue balls. Each set of balls is numbered 1 to 10. Suppose 7 balls are drawn at random from the box. In how many ways can there be 3 of one colour, 2 of a second colour and 2 of the 3rd colour. ■

Exercise 2.8 How many paths of length 2 edges are in K_6 ? ■

2.5 Counting in Graphs

2.5.1 Lecture 5 (Grimaldi 11.1, 11.3)

Bipartite graphs

Example 2.20 Draw the graph $G = (V,E)$ where $V = \{1,2,3,4,5\}$ and $E = \{\{1,3\},\{1,4\},\{2,4\},\{2,5\}\}$.

Definition 2.5.1 — Bipartite graph. A graph $G = (V,E)$ is **bipartite** if we can partition the vertices into two nonempty sets V_1 and V_2 such that:

(1) $V_1 \cap V_2 = \emptyset$

(2) $V_1 \cup V_2 = V$

(3) every edge in E is incident with one vertex in V_1 and one vertex in V_2.

Definition 2.5.2 — $K_{m,n}$. For integers $n \geq 1$ and $m \geq 1$ we define the **complete bipartite graph** $K_{m,n}$ to be a bipartite graph with $|V_1| = m$, $|V_2| = n$ and

$$E = \{\{v_1, v_2\} \mid v_1 \in V_1 \text{ and } v_2 \in V_2\}.$$

Example 2.21 $K_{2,3}$

Example 2.22
Question 1: How many edges are in a path on n vertices?
Question 2: How many edges are in a cycle on n vertices?

Question 3: How many edges are in K_n?
Question 4: How many edges are in K_{n_1,n_2}?

Example 2.23
Question 5: How many graphs are there with n vertices?
Question 6: How many graphs are there with n vertices and m edges?

Let V_1, V_2 be disjoint sets with $|V_1| = n_1$ and $|V_2| = n_2$.
Question 7: How many graphs have bipartition (V_1, V_2)?
Question 8: How many graphs have bipartition (V_1, V_2) with m edges?

Subgraphs

Definition 2.5.3 — Subgraph. Let $G = (V, E)$ and $G' = (V', E')$ be two graphs.
G' is a **subgraph** of G if $V' \subseteq V$ and $E' \subseteq E$.
If $V' = V$ then we call G' a **spanning** subgraph of G.

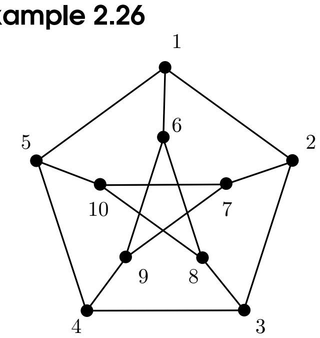
Example 2.24

Example 2.25
Question 9: How many spanning subgraphs does K_{n_1,n_2} have?
Question 10: How many spanning subgraphs of K_{n_1,n_2} have exactly m edges?

Paths and cycles

Definition 2.5.4 — Paths and Cycles.
If P is a subgraph of G that is a path we call P **a path of** G.
If C is a subgraph of G that is a cycle we call C **a cycle of** G.

Example 2.26



The Petersen graph.

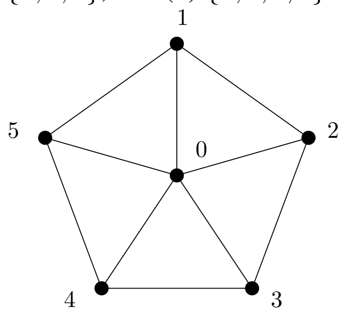
Example 2.27
Question 11: How many 4-vertex paths does the graph K_n have?

Induced subgraphs

Definition 2.5.5 — Induced subgraph. Let $G = (V,E)$ be a graph and let $V' \subseteq V$. The subgraph of G **induced** by V' is the subgraph $G' = (V',E')$ where

$$E' = \{\{x,y\} \mid x \in V, y \in V' \text{ and } \{x,y\} \in E\}.$$

Example 2.28 For the graph below determine the induced subgraph for the vertex sets (a) $\{1,3,4\}$, and (b) $\{0,1,3,4\}$.



Exercise 2.9 If $G = (V,E)$ is a graph with $|V| = n$, how many induced subgraphs does G have?

2.5.2 Additional Comments

Here we provide another detailed solution to the second question in Example 2.27.

Problem: How many k-vertex paths does K_{n_1,n_2} have?

Solution: Assume that our K_{n_1,n_2} has bipartition $(V1,V2)$ with $|V_i| = n_i$ for $i = 1,2$.We will select a path by choosing a sequence of distinct vertices $(v_1, v_2, v_3, ..., v_k)$ where the vertices alternate between the two sets in our bipartition. This sequence determines the path with vertex set $\{v_1,\ldots,v_k\}$ and edge set $\{\{v_1,v_2\},\ldots,\{v_{k-1},v_k\}\}$. Note that this same path is also selected by the "opposite" vertex sequence $(v_k, v_{k-1},\ldots,v_1)$.

Case 1: k **even.** Suppose $k = 2t$. In this case our path will start and end in different sides of the bipartition. So in this case, the total number of paths of length k will be exactly equal to the number of sequences of distinct vertices $(v_1, v_2, v_3, \ldots, v_k)$ where the vertices alternate between the two sets in our bipartition and we start with $v_1 \in V_1$. (I.e. this counts each path exactly once.) Here we have n_1 choices for the first vertex, n_2 choices for the second, then $n_1 - 1$ for the third, $n_2 - 1$ for the fourth, and so on. Therefore, for $k = 2t$ the total number of paths is given by the formula:

$$n_1(n_1-1)(n_1-2)\cdots(n_1-t+1)n_2(n_2-1)(n_2-2)\cdots(n_2-t+1) = \frac{n_1!n_2!}{(n_1-t)!(n_2-t)!}$$

Case 2: k **odd.** Suppose $k = 2t+1$. In this case our path will either start and end in V_1 or start and end in V_2. First suppose it starts and ends in $V-1$. As before, we will be selecting a sequence of distinct vertices $(v_1, v_2, \ldots, v_k)$ that alternate between the sides with $v_1 \in V_1$. The number of these sequences is given by:

$$n_1(n_1-1)(n_1-2)\cdots(n_1-t)n_2(n_2-1)(n_2-2)\cdots(n_2-t+1) = \frac{n_1!n_2!}{(n_1-t-1)!(n_2-t)!}$$

However, now the sequences $(v_1, v_2, \ldots, v_{k-1}, v_k)$ and $(v_k, v_{k-1}, \ldots, v_2, v_1)$ both determine the same path. So we will need to divide by 2 to get the correct amount. This gives us a total of

$$\frac{n_1!n_2!}{2(n_1-t-1)!(n_2-t)!}$$

paths starting and ending in V_1. A similar analysis shows that there are

$$\frac{n_1!n_2!}{2(n_1-t)!(n_2-t-1)!}$$

paths starting and ending in V_2. Summing these gives the total number of K-vertex paths in our graph K_{n_1,n_2}

$$\frac{n_1!n_2!}{2(n_1-t-1)!(n_2-t)!} + \frac{n_1!n_2!}{2(n_1-t)!(n_2-t-1)!}$$

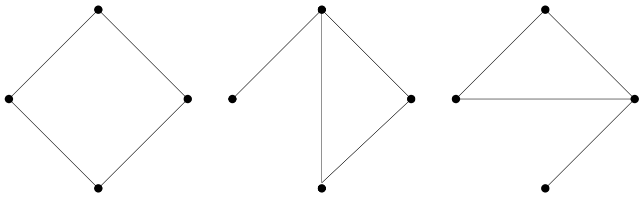
■

2.6 Graph Isomorphism

2.6.1 Lecture 6 (Grimaldi 11.2)

Isomorphism

Example 2.29 Which of the following graphs are the 'same'?



isomorphism: the word derives from the Greek *iso*, meaning "equal" and *morphosis*, meaning "form".

Definition 2.6.1 — Isomorphic graphs. Let $G = (V_1, E_1)$ and $H = (V_2, E_2)$ be two graphs. Then G is **isomorphic** to H (has the same structure as) if there is a bijection $f : V_1 \to V_2$ such that:

$$\{u, v\} \in E_1 \quad \Leftrightarrow \quad \{f(u), f(v)\} \in E_2.$$

The function f is called an **isomorphism**.

Note: (i) G_1 and G_2 are isomorphic if the same drawing is valid for both graphs (but with different labels on the vertices).
(ii) By the definition, any two paths on the same number of vertices are isomorphic. The same is true for any two cycles on the same number of vertices.

Example 2.30

Example 2.31 Draw all non-isomorphic graphs with $|V| = 3$ and $|V| = 4$.

Exercise 2.10 Draw all non-isomorphic graphs with 5 vertices and 4 edges. ■

How can we test if two graphs G and H are isomorphic?

An "efficient" graph isomorphism algorithm is not known.

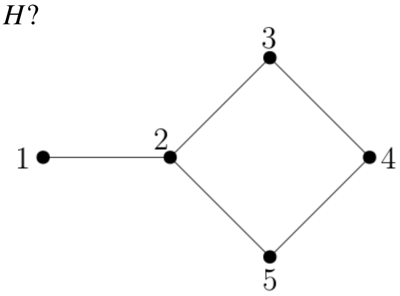
Counting subgraphs

Example 2.32 For $n \geq t$, how many subgraphs of K_n are isomorphic to K_t?

Example 2.33 Let K_4^- be the graph obtained from K_4 by deleting an edge. How many subgraphs of K_n are isomorphic to K_4^-?

Example 2.34 How many subgraphs of $K_{n,m}$ are isomorphic to $K_{3,4}$?

Example 2.35 Let H be the graph shown below. How many subgraphs of $K_{n,n}$ are isomorphic to H?



2.7 Summary

Sequences: The notion of a sequence corresponds to a totally ordered set of objects from a given alphabet: there is an object in the first position, and object in the second position, ... A *permutation* is a sequence with no repetitions (each alphabet object can appear at most once).
We consider two kinds of alphabets. First, if an alphabet of size n is composed of distinct objects:

- The number of sequences of size r is n^r.
- The number of permutation of size r is $\dfrac{n!}{(n-r)!}$.

Next, if an alphabet is a multiset with repeated objects (n_i objects of type i, with k different types of objects and $n = n_1 + \cdots + n_k$), we saw that:

- The number of permutations of size n (i.e. each object appears exactly once) is

$$\frac{n!}{n_1! \cdots n_k!}$$

All these results follow from the following principle: a family of sequences defined by some characteristics (composition, prescribed positions, ...) reduces to figure out the number of choices for the first position, then for the second position, and so on, and then these values are multiplied together.

Subsets: Next, when order doesn't matter, we are dealing with subsets of a ground set of size n that has only distinct objects.

- The number of subsets/combinations of size r with no repetition is the binomial number $C(n,r) = \binom{n}{r} = \frac{n!}{r!(n-r)!}$.
- The number of ways to choose a collection of size r (from the set of size n) with possible repetitions is $\binom{n+r-1}{r}$.
- The number $\binom{n+r-1}{r}$ is also the number of positive integer solutions to $x_1 + x_2 + \cdots x_n = r$, as well as the number of ways to place r identical objects into n distinct bins.

The Binomial Theorem: $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$
It is important to understand the explanation in terms of counting sequences of length n over an alphabet composed of two disjoint sets of x and y objects, respectively.

Vocabulary

- sequence, string, word, alphabet, ground set
- permutations
- subsets, combinations, multiset
- Binomial numbers
- graph, walk, path, cycle, complete graph, bipartite graph, graph isomorphism

Skills to acquire

- Master the vocabulary of sequences, subsets and graphs.
- Count families of sequences/combinations over a given alphabet described by characteristics in terms of content, prescribed positions, ...
- Understand the binomial theorem and its proof.
- Understand the proof for combinations with repetitions (stars and bars)