MATH 510: Discrete Mathematics Notes

Kale Stahl

Summer 2021

Contents

1 Graph Theory

1.1 Graph Models

A graph, G, is notated as $G = (V, E)$ where V is a set of vertices and E is a set of edges. A vertex is a given point, and an edge is defined a segment which connects two vertices. If an edge connects vertices a and b, it can be notated as (a, b). Given an edge $(a, b) \in E$, we say that a is **adjacent** to b.

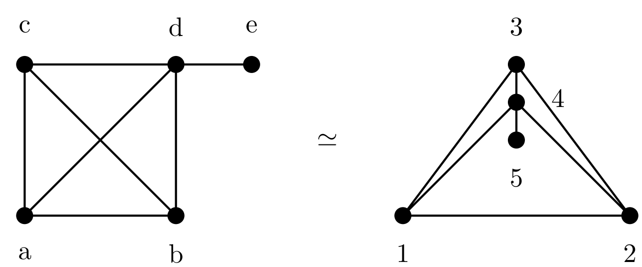
A **directed graph** is a graph in which the ordering of the edges matter. Directed graphs are visually represented as and are notated usually as $(\vec{a}, b)$. Note that $(\vec{a}, b)$ is not the same as $(\vec{b}, a)$.

A **path** is a sequence of connected vertices written as $P = x_1 - x_2 - \cdots - x_n$. A path is a **circuit** if it also includes the edge (x_n, x_1). A path is **connected** if there exists a path between any two vertices.

A graph G is **bipartite** if it can be divided into two sets of vertices such that all edges connect a vertex from the first set to a vertex in the other set. The number of edges including a given vertex is the **degree** of the vertex.

1.2 Isomorphism

Two graphs G and G' are **isomorphic** if there exists a 1 to 1 correspondence between vertices in both G and G' such that a pair of vertices in G is adjacent if and only if the corresponding pair of vertices is adjacent in G'. Note that two graphs can only be isomorphic if they have the same number of edges and vertices. The graphs shown below are isomorphic.



A **subgraph** of a graph is a graph formed by a subset of the vertices and edges of the original graph. If the subgraphs of two graphs are isomorphic, then the original graphs are also isomorphic.

A graph is **complete** when there exists an edge between all distinct pairs of vertices. A complete graph with n vertices is notated as K_n. K_1 is a point, K_2 is an edge, and K_3 is often called a triangle.

Given a graph $G = (V, E)$, its **complement** is $\bar{G} = (V, \bar{E})$. $\bar{G}$ has the same vertices as G, but only includes edges that are not present in G. If G is isomorphic, then $\bar{G}$ is also isomorphic. A subgraph of mutually nonadjacent vertices is called a set of **isolated vertices**.

1.3 Edge Counting

For directed graphs, the **in-degree** of a vertex is the number of edges pointing toward the vertex while the **out-degree** is the number of edges pointing away from the vertex. If the in-degree and out-degree of the vertex are different, then the vertex is **mixed degree**.

Theorem 1. $\forall G, \sum_{i=1}^{n} \deg v_i = 2e$

Proof. Summing all degrees of each a vertex counts all instances of some edge being incident to a vertex. But each edge is incident to two vertices, os each edge will be counted twice. $\square$

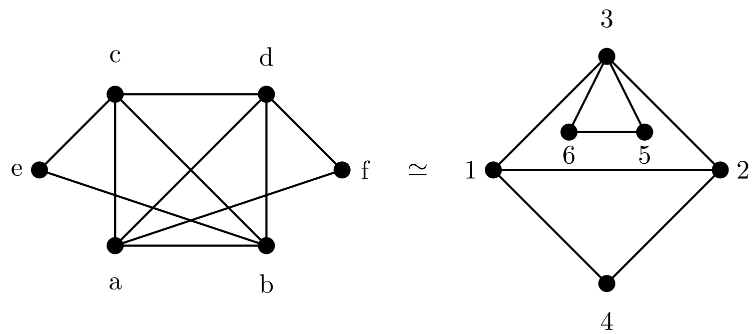
Corollary 1.1. *The number of vertices with an odd degree in any graph G is always even.*

If a graph G is not connected, its vertices can be partitioned into connect pieces called **components**. Formally, a component H is a connected subgraph of G such that there is no path between a vertex in H and any vertex of G not in H. Note that Corollary 1.1 also holds for components.

Theorem 2. *A graph G is bipartite if and only if every circuit of G has an even length.*

1.4 Planar Graphs

We say that a graph G is **planar** if it can be drawn such that no edges cross. A **plane graph** is a planar depiction of a planar graph. The following graphs are isomorphic and planar



To show that a graph is planar, the most common method is the **circle chord method**. The method is:

A. Find a circuit containing all vertices in a graph. Draw this circuit as a circle.

B. Fill in the rest of the edges not contained in the circuit as "chords". These cords can be either inside or outside of the circle.

C. Continue filling in chords such that they do not cross. If there is no way to draw a chord such that it does not cross any edges, then the graph is nonplanar. If all edges can be drawn, the graph is planar.

Theorem 3 (Euler's Theorem)**.** *If G is connected and planar, then a planar depiction has a number of regions equal to $e - v + 2$. It is assumed that every graph has 1 unbounded region on the outside of the graph.*

Corollary 3.1. *Assume G is a planar graph with more than one edge. Then it can be said of G that $e \leq 3v - 6$.*

The **degree of a region** is the number of edges on the boundary of a region. If a graph is bipartite, there can be no regions of degree 3 or less, every region must have a degree of at least 4. The improved bound from corollary 3.1 for bipartite graphs is $e \leq 2v - 4$.

2 Circuits and Graph Coloring

2.1 Euler Cycles

A **multigraph** is a generalized graph that allows for multiple edges between vertices and loops of the form (a, a). A **trail** is a sequences of vertices joined by an edge but has no repeated edges. A **cycle** is a circuit but all edges are distinct.

Theorem 4. *An undirected multigraph has an Euler Cycle if and only if it is connected and all of its vertices have an even degree.*

Proof.

- ⇒ Suppose a multigraph G is connected and has all vertices of an even degree. Pick any vertex a and trace a trail. The even degree condition ensures that we will not stop at another vertex and therefore must return to a Let C be the cycle generated from this path and let G' be the multigraph consisting of the edges of $G \backslash \{C\}$ Since G is connected, C and G' must have a common vertex or else there is no path from vertices in G'. Let a' be this vertex. Now build a cycle C' through G' from a'. Incorporate C' into C at a'. Repeat this process until there are no remaining edges.

- ⇐ Suppose G has an Euler Cycle C. Then G is connected since C is connected. Since C connects all vertices, we leave each vertex just as often as we arrive at each vertex, each vertex has an even degree.

□

An **Euler trail** is a trail that contains all edges in G and visits each vertex exactly once. If you remove an edge from a graph with an Euler cycle, there will no longer be an Euler cycle in the graph.

Corollary 4.1. *A multigraph has an Euler trail but not an Euler cycle if and only if it is connected and has exactly 2 vertices of an odd degree.*

2.2 Hamilton Circuits

Hamilton circuits and **Hamilton path** is a circuit that visits each vertex exactly once. There are 3 rules that every set of edges must satisfy in order to form a Hamilton circuit. These rules are:

A. If a vertex x has degree 2, both of the edges incident to x must be part of any Hamilton circuit.

B. No proper subcircuit, that is, a circuit containing all vertices, can be formed when building a Hamilton circuit.

C. Once the Hamilton circuit is required to use two edges at a vertex x, all other (unused) edges incident at x must be removed from consideration.

Note that any Hamilton circuit must contain exactly 2 edges incident to each vertex.

Theorem 5. *A graph with n vertices, $n > 2$, has a Hamilton circuit if the degree of each vertex is at least $\frac{n}{2}$.*

Theorem 6. *Let G be a connected graph with n vertices, and let the vertices be indexed $x_1, x_2, \ldots, x_n$, so that* $\deg(x_i) \leq \deg(x_i + 1)$. *If for each $k \leq \frac{n}{2}$, either* $\deg(x_k) > k$ *or* $\deg(x_n - k) \geq n - k$, *then G has a Hamilton circuit.*

Theorem 7. *Suppose a planar graph G has a Hamilton circuit H. Let G be drawn with any planar depiction, and let r_i denote the number of regions inside the Hamilton circuit bounded by i edges in this depiction. Let r'_i be the number of regions outside the circuit bounded by i edges. then the numbers r_i and r'_i satisfy the equation*

$$\sum_i (i-2)(r_i - r'_i) = 0$$

A **tournament** is a directed graph obtained from a complete (undirected) graph by giving a direction to each edge.

Theorem 8. *Every tournament has a Hamilton path.*

2.3 Graph Coloring

A **chromatic number** is the minimum number of colors needed to color a graph. The chromatic number of a graph G is notation as $\chi(G) = K$. The **chromatic polynomial** is a polynomial in K (the number of colors). It reduces to the number of possible ways to K-color a graph. A chromatic polynomial is denoted by $P_K(G)$. If K is small so that G is not K-colorable, then we say that $P_K(G) = 0$. A coloring of a graph G assigns K colors to the vertices of G. To prove that the chromatic number of G is K, we need to show:

A. A K-coloring exists.

B. That $K-1$ colors don't suffice.

Some general rules for graph coloring are:

A. K_n requires n colors and cannot be $n-1$-colored.

B. To build a K-coloring, you ignore vertices of degree less than K to start.

When looking at a coloring, do not simply look to the largest complete subgraph. A 5-spoke wheel is an example of why not to do this. In general, for a wheel W with n spokes,

$$\chi(W) = \begin{cases} 3 & \text{for } n \text{ odd} \\ 4 & \text{for } n \text{ even} \end{cases}$$

2.4 Coloring Theorems

A **polygon** is any single circuit with edges drawn as straight lines. The **triangulation of a polygon** is the act of drawing straight, non-crossing chords between vertices of a polygon so that all regions created are triangles

Theorem 9. *The vertices of a triangulation of a polygon can be 3-colored. Note: the polygon does not necessarily need to be convex.*

Corollary 9.1. *An art gallery problem with n vertices will require, at most, $\lfloor \frac{n}{3} \rfloor$ guards.*

Corollary 9.2. *A graph is bipartite if and only if it is 2-colorable. This also implies that a graph is 2-colorable if and only if all circuits in the graph are even.*

Theorem 10. *If G is not an odd circuit or a complete graph, $\chi(G) \leq d$ where d is the maximum degree of a vertex.*

An **edge coloring** of a graph is the act of coloring edges such that any 2 edges with a common vertex have different colors.

Theorem 11. *If the maximum degree of a vertex in a graph G is d, the the edge chromatic number of G is either d or $d+1$.*

Theorem 12. *Every planar graph can be 5-colored.*

2.5 Instant Insanity Puzzle

In this puzzle, there are 4 cubes, each with one of 4 colors on each face. The goal is to stack the cubes vertically so that any side of the stack has each color appearing exactly once. Trying to brute force the puzzle proves fruitless, as there are $24^4 = 331,776$ possible combinations. We notice that you can "lock" two opposite faces on the cube and then rotate the remaining 4 faces without changing the locked faces. From this, we can create see that there are actually disjoint problems in this puzzle. This is illustrated in the decomposition principle.

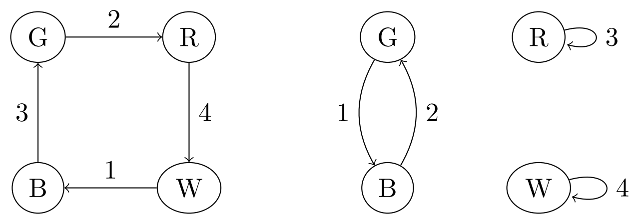
Decomposition Principle

A. Pick one pair of opposite faces on each cube for the left and right sides of the pile so that these two sides of the pile will have one face of each color.

B. Pick a different pair of opposite faces on each cube for the front and back sides of the pile so that these two sides will have one face of each.

Now that we have simplified the problem, we can now restate it in graph theoretic terms. Via the decomposition principle, we can break the puzzle into a left-right part and a front-back part. We can start with either part by drawing a multigraph with one vertex for each color, and then drawing an edge between colors that can appear on opposite sides and label each each with the order of the cube in the pile.

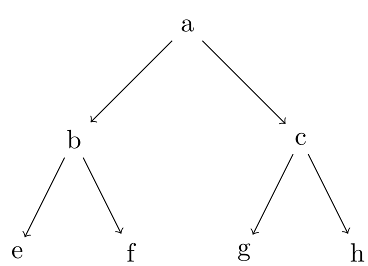
When solving, we want to look for a **factor** of our multigraph. In an n-vertex multigraph, a set of n edges forming disjoint simple circuits is a factor. Note that a factor is a generalization of a Hamilton circuit. In our puzzle, let a **labeled factor** be a factor in which each number appears once. In order to solve this puzzle, we want to find two distinct labeled factors within our labeled graph. An example of these two labeled factors would be:



3 Trees and Searching

3.1 Properties of trees

In undirected graphs, a **tree** is a connected graph with no circuits. More generally, a tree is a connected graph with a designated vertex called a **root** such that there is a unique path from the root to any other vertex in a tree. A **rooted tree** is a directed tree in there are unique roots where there are no circuits between them. The following graph is a rooted tree with a root of a:



The **level number** of a vertex is the length of a unique path from the root to that vertex. For example, in the tree above, e has a level number of 2. For any vertex x in a rooted tree T, except the root, a **parent** of x is the vertex y with an edge $\overrightarrow{(y, x)}$ into x. The **children** of x are vertices z with an edge directed from x to z. All children of x will have a level number one greater than x. Two vertices with the same parent are **siblings**.

Theorem 13. *A tree with n vertices has $n - 1$ edges.*

Vertices of T with no children are called **leaves** of T. Vertices with children are called **internal vertices**. If all internal vertices of a rooted tree has m children, then it is an **m-ary tree**.

Theorem 14. *Let T be an m-ary tree with n vertices, of which i vertices are internal. Then, $n = mi + 1$.*

Corollary 14.1. *Let T be an m-ary tree with n vertices consisting of i internal vertices and l leaves. If we know one of n, i, or l, then the other two parameters are given by the following formulas:*

(a) Given i, then $l = (m - 1)i + 1$ and $n = mi + 1$.

(b) Given l, then $i = \frac{(l-1)}{(m-1)}$ and $n = \frac{(ml-1)}{(m-1)}$.

(c) *Given n, then $i = \frac{(n-1)}{m}$ and $l = \frac{[(m-1)+1]}{m}$.*

Theorem 15. *Let T be an m-ary tree of height h with l leaves. Then;*

(a) *$l \leq m^h$, and if all leaves are at height h, $l = m^h$.*

(b) *$h \geq \lceil \log_m l \rceil$, and if tree is balanced, $h = \lceil \log_m l \rceil$*

Theorem 16. *There are n^{n-2} different undirected trees on n labels.*

3.2 Search Trees and Spanning Trees

A **spanning tree** of a graph G is a subgraph of G that is a tree containing all vertices of G. Spanning trees can be built breadth-first or depth-first.

To build a depth-first spanning tree, we pick some vertex as the root and begin building a path from the root composed of edges of the graph. This path continues until it can no longer go further without repeating a vertex. We then backtrack until we can start and new path and continue until all vertices of G are used.

To build a breadth-first spanning tree, we pick some vertex x as the root and put all edges leaving x (along with the vertex they connect x to) in the tree. Then we successively add to the tree the edges leaving the vertices adjacent to x, and then continue level by level until all vertices are in the tree.

It is important to note that if a G is not connected, then no spanning tree can exist. This makes spanning trees a simple way to test a graph for connectedness. Another way to test for connectedness, is an **adjacency matrix**. This is a (0,1)-matrix where a 1 in (i, j) if vertex x_i and x_j are connected and 0 otherwise. An adjacency matrix for a graph could look like:

	x_1	x_2	x_3	x_4
x_1	0	1	1	0
x_2	1	0	1	1
x_3	1	1	0	1
x_4	0	1	1	0

Note that if an adjacency matrix is not symmetric, then it is a directed graph.

3.3 The Traveling Salesman Problem

We can use trees in order to solve the famous "Traveling Salesman Problem" by optimizing a cost matrix associated with it. To optimize the traveling salesman problem, we are looking for a labeled Hamilton circuit in which the sum of the costs of each edge is minimal.

4 General Combinatorics

4.1 Basic Counting Principles

There are two fundamental principles in counting:

Addition Principle

If there are r_1 different objects in the first set, r_2 different objects in the second set, and r_m different objects in the m^{th} set,..., and *if the different sets are disjoint*, then the number of ways to select an object from one of the m sets is $r_1 + r_2 + \cdots + r_m$.

Multiplication Principle

Suppose a procedure can be broken into m successive (ordered) stages, with r_1 different outcomes in the first stage, r_2 different outcomes in the second stage, ..., and r_m different outcomes in the m^{th}. If the number of outcomes at each stage is independent of the choices in the previous stages and *if the composite outcomes are all distinct*, then the total procedure has $r_1 \times r_2 \times \ldots \times r_m$ different composite outcomes.

4.2 Simple Arrangements and Selections

A **permutation** of n distinct objects is an arrangement, or ordering, of n objects. An r-**permutation** of n distinct objects is an arrangement using r of the n objects. An r-**combination** of n distinct objects is an unordered selection, or subset, of r out o fthe n objects. $P(n,r)$ denotes the r-permutation of n objects and $C(n,r)$ denotes the r-combination of n objects. Note that

$$P(n,n) = n!$$

and

$$P(n,r) = n(n-1)(n-2) \times \ldots \times [n-(r-1)] = \frac{n!}{(n-r)!}$$

due to the application of the multiplication principle. $C(n,r)$ is sometimes referred to as **binomial coefficients** due to their role in the expansion of $(x+y)^n$. Note that $C(n,r)$ can be written as $\binom{n}{r}$ and is read "n choose r". We can calculate $C(n,r)$ by picking an $r$permutation of n and dividing it by an n-permutation of n objects. This is written as

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

4.3 Arrangements and Selections with Repetition

When finding arrangements with repetition, the key is to focus on the subset of positions where the repeated elements go. So if we were to find the number of ways to arrange the letters in "banana", we would first find where each "a" went then focus on where each "n" can go after the "a"s have been placed. So, there are $C(6,3) = 20$ ways to place the "a"s. Then there are $C(3,2) = 3$ ways to place the remaining "n"s and $C(1,1) = 1$ ways to place the 'b'. Then, by the multiplication principle, there are 60 possible arrangements.

Theorem 17. *If there are n objects, with r_1 of type 1, r_2 of type 2, ..., and r_m of type m, where $r_1 + r_2 + \cdots + r_m = n$, then the number of arrangements of these n objects, denoted $P(n; r_1, r_2, \ldots, r_m)$, is*

$$P(n; r_1, r_2, \ldots, r_m) = \binom{n}{r_1}\binom{n-r_1}{r_2}\binom{n-r_1-r_2}{r_3}\cdots\binom{n-r_1-r_2\cdots-r_{m-1}}{r_m} = \frac{n!}{r_1!r_2!\ldots r_m!}$$

Theorem 18. *The number of selections with repetition of r objects chosen from n types of objects is $C(r+n-1,r)$.*

4.4 Distributions

General guidelines for modeling distribution problems are:

Distributions of objects are equivalent to arrangements
and
Distributions of identical objects are equivalent to selections

Basic Models for Distributions

Distinct Objects The process of distributing r distinct objects into n different containers is equivalent to putting the distinct objects in a row and stamping one of the n different box names on each object. the resulting sequence of box names is an arrangement of length r formed from n items (box names) with repetition. Thus, there are $n \cdot n \cdot, \ldots, \cdot n = n^r$ distributions of the r objects. If r_i objects must go in box i, $1 \leq i \leq n$, then there are $P(r; r_1, r_2, \ldots, r_n)$ distributions.

Identical Objects The process of distributing r identical objects into n different boxes is equivalent to choosing an (unordered) subset of r box names with repetition from among the n choices of boxes. Thus, there are $C(r+n-1, r) = \frac{(r+n-1)!}{r!(n-1)!}$ distributions of the r identical objects.

The following table summarizes the different basic types of counting problems with distributions:

	Distribution of Distinct Objects	Distribution of Identical Objects
No Repetition	$P(n, r)$	$C(n, r)$
Unlimited Repetition	n^r	$C(n+r-1, r)$
Restricted Repetition	$P(n; r_1, r_2, \ldots r_m)$	—

4.5 Binomial Identities

A **binomial** is a sum of two monomials as in, $c_1 x^{k_1} + c_2 x^{k_2}$ for $k_1, k_2 \geq 0$. The following are multiple identities relating to the binomial coefficient discussed earlier.

Theorem 19 (Binomial Theorem)**.** $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$

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

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1} \tag{2}$$

$$\binom{n}{k}\binom{k}{m} = \binom{n}{m}\binom{n-m}{k-m} \tag{3}$$

$$k\binom{n}{k} = n\binom{n-1}{k-1} \qquad \text{or} \qquad \binom{n}{k} = \frac{n}{k}\binom{n-1}{k-1} \tag{4}$$

$$\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n} = 2^n \tag{5}$$

$$\binom{n}{0} + \binom{n+1}{1} + \binom{n+2}{2} + \cdots + \binom{n+r}{r} = \binom{n+r+1}{r} \tag{6}$$

$$\binom{r}{r} + \binom{r+1}{r} + \binom{r+2}{r} + \cdots + \binom{n}{r} = \binom{n+1}{r+1} \tag{7}$$

$$\binom{n}{0}^2 + \binom{n}{1}^2 + \binom{n}{2}^2 + \cdots + \binom{n}{n}^2 = \binom{2n}{n} \tag{8}$$

$$\sum_{k=0}^{r} \binom{m}{k}\binom{n}{r-k} = \binom{m+n}{r} \tag{9}$$

$$\sum_{k=0}^{m} \binom{m}{k}\binom{n}{r+k} = \binom{m+n}{m+r} \tag{10}$$

$$\sum_{k=0}^{m} \binom{m-k}{r}\binom{n+k}{s} = \binom{m+n+1}{r+s+1} \tag{11}$$

5 Recurrence Relations

5.1 Recurrence Relation Models

A **recurrence relation** is a recursive formula that counts the number of ways to do a procedure involving n objects in terms of fewer objects. For example, assume a_k is the number of ways to do a procedure with k objects for $k \in \mathbb{Z}_{\geq 0}$, then its recurrence relation is an equation that expresses a_n in terms of preceding values a_k for $k < n$. A simple model for a_n could be $a_n = 2a_{n-1}$.

A recurrence relation needs **initial conditions** in order to correctly compute the relation. These are similar to the initial conditions needed when solving differential equations. An initial condition for the example recurrence relation could be $a_0 = 1$. These are usually the easiest cases to calculate logically for the relation, and since the relation is recursive, as long as we have the first few cases, we can calculate all subsequent cases.

5.2 Divide and Conquer Relations

The main goal for divide and conquer relations is to split the problem into two sub-problems of the the size. These are useful for solving problems involving trees. The total number of steps a_n required by a divide and conquer algorithm to process n elements most likely satisfies the recurrence relation

$$a_n = ca_{n/2} + f(n)$$

where c is a constant and $f(n)$ is a function of n. We can solve a recurrence relation to remove the recursive element and make a_n only a function of n. To do this, we first determine the recurrence relation for the problem. Then to solve, we look at the following table to determine what form the solution will take:

c	$f(n)$	a_n
$c = 1$	d	$d\lceil \log_2 n \rceil + A$
$c = 2$	d	$An - d$
$c > 2$	dn	$An^{\log_2 c} + \left(\frac{2d}{2-c}\right) n$
$c = 2$	dn	$dn(\lceil \log_2 n \rceil + A)$

The constant A is to be chosen to fit the initial conditions of the relation.

5.3 Linear Recurrence Relations

A recurrence relation is linear if it is written in the form

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_r a_{n-r}$$

where the c_is are constants. Solving these are very similar to solving linear differential equations. We know that the solution will be a linear combination of the form $a_n = \alpha^n$. So then we can begin by substituting for a_n and then solving for α. Substituting α^n for a_n we find

$$\alpha^n = c_1 \alpha^{n-1} + c_2 \alpha^{n-2} + \cdots + c_r \alpha^{n-r}$$

We can then divide both sides by α^{n-r} to get

$$\alpha^r = c_1 \alpha^{r-1} + c_2 \alpha^{r-2} + \cdots + c_r$$

and we can rearrange the equation to get the form of a polynomial

$$\alpha^r - c_1 \alpha^{r-1} - c_2 \alpha^{r-2} - \cdots - c_r = 0$$

This polynomial is called the **characteristic equation**. This characteristic polynomial has r roots $\alpha_1, \alpha_2, \ldots, \alpha_r$. Then for any i, $0 \leq i \leq r$, $a_n = \alpha_i^n$ is a solution to the equation. The whole solution can be expressed as a linear combination of solutions in the form

$$a_n = A_1 \alpha_1^n + A_2 \alpha_2^n + \cdots + A_r \alpha_r^n$$

where the A_is are constants. We can solve for A_i by using initial conditions and then solving the system of equations that are created.

5.4 Inhomogeneous Recurrence Relations

A recurrence relation is considered **homogeneous** if all of the terms of the relation involve some form of a_k. A homogeneous recurrence relation will look like: $c_1a_n + c_2a_{n-2} + \cdots + c_ra_{n-r} = 0$. An **inhomogeneous** recurrence relation looks similar, except contains a function of n, as in $a_n = ca_{n-1} + f(n)$.

Solving inhomogeneous recurrence relations is similar to that of ordinary differential equations. We will first solve the homogeneous part of the problem where $f(n) = 0$, and then find the solution to the inhomogeneous part, called the particular solution. When finding the particular solution, we can use the following table where d and e are constants:

$f(n)$	Particular solution $p(n)$
d	B
dn	$B_1n + B_0$
dn^2	$B_2n^2 + B_1n + B_0$
ed^n	Bd^n

The constants B are to be determined when solving. Note that if $f(n)$ were a sum of several different terms, we would solve each term separately and then add them together to find the whole particular solution. However, this will not work for the case of $a_n = da_{n-1} + ed^n$. For this, try $a_n^* = Bnd^n$ as the particular solution.

6 Inclusion-Exclusion

6.1 Counting with Venn Diagrams

We will go over some notation we will use in this section. Let $N(S)$ be the number of elements in set S. We will let $N = N(U)$, the universal set. The complement of a set A, $\bar{A}$, is the set of all elements that are not in A. So, $N(A) = N - N(\bar{A})$, or $N(\bar{A}) = N - N(A)$.

When we find $N(A \cup B)$ we are overcounting $N(A \cap B)$. So then to find the number of elements in $A \cup B$, we need to add the sets and subtract what we overcounted, so then $N(A \cup B) = N(A) + N(B) - N(A \cap B)$. We will ise this idea of overcounting and then subtracting the elements in which we counted more than once more throughout this section. A good strategy for counting objects is to create sets in which we want to find the complement of the set. Note that $A_1A_2 \ldots A_n$ is often used as an equivalent to $A_1 \cap A_2 \cap \cdots \cap A_n$

6.2 Inclusion-Exclusion Formula

We will now generalize the overcounting principle we saw previously as:

Theorem 20 (Inclusion-Exclusion Formula)**.** *Let $A_1, A_2, \ldots A_n$, be an n sets be in a universe U of N elements. Let S_k denote the sum of the sizes of k-tuple intersections of the A_is. Then,*

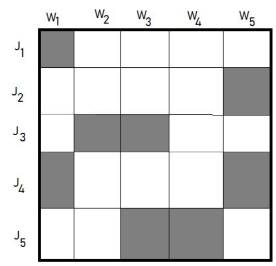
$$N(\bar{A}_1\bar{A}_2 \ldots \bar{A}_n) = N - S_1 + S_2 - S_3 + \cdots + (-1)^k S_k + \cdots + (-1)^n S_n$$

Corollary 20.1. *Let $A_1, A_2, \ldots A_n$ be sets in the universe U. Then,*

$$N(A_1 \cup A_2 \cup \cdots \cup A_n) = S_1 - S_2 + S - 3 - \cdots + (-1)^{k-1} S_k + \cdots + (-1)^{n-1} S_n$$

6.3 Restricted Positions and Rook Polynomials

A popular use of the inclusion-exclusion formula is in the problem of restricted positions on a board. For this example, we will have a two-dimensional grid in which squares that are darkened are forbidden. So then in the following grid, let W_i be the ith worker and J_i be the ith job. If a square is darkened, then that worker is not qualified for that job, and therefore cannot be placed there.



This problem is most often generalized as rooks on a chessboard where certain spaces are restricted. The goal is to find how many arrangements of n rooks can be placed on the board such that no rook captures another. To do this, we can rearrange the rows and columns on the board without changing any possible placements. We will do this in order to create two disjoint subboards B_1 and B_2 such that each subboard has its own set of rows and columns. We will define $r_k(B)$ to be the number of ways to place k noncapturing rooks on board B.

Lemma 21. *If B is a board of darkened squares that decomposes into tow disjoint sub-boards B_1 and B_2, then,*

$$r_k(B) = r_k(B_1)R_0(B-2) + r_{k-1}(B_1)r_1(B_2) + \cdots + r_0(B_1)r_k(B_2)$$

A **rook polynomial** $R(x, B)$ of the board B of darkened squares is defined as

$$R(x, B) = r_0(B) + r_1(B)x + r_2(B)x^2 + \ldots$$

Note that $r_0(B) = 1$ for all B. The rook polynomial only depends on the darkened squares, not on the size of the original board.

Theorem 22. *If B is a board of darkened squares that decomposes into the two disjoint subboards B_1 and B_2, then*

$$R(x, B) = R(x, B_1)R(x, B_2)$$

.

Theorem 23. *The number of ways to arrange n distinct objects when there are restricted positions is equal to*

$$n! - R_1(B)(n-1)! + r_2(B)(n-2)! + \cdots + (-1)^k r_k(B)(n-k)! + \cdots + (-1)^n r_n(B)0!$$

A general series of steps when solving restricted position problems is:

1. Create an array counting arrangements or matches with restricted positions by darkening squares of forbidden matches.
2. Rearrange the rows and columns in the array to create disjoint subboards.
3. Determine the $r_k(B_i)$ for each subboard i.
4. Use $r_k(B_i)$ to form the rook polynomials $R(x, B_i)$ and apply Theorem 22.
5. Use the rook polynomial to calculate the expression in Theorem 23.

7 Pidgeonhole Principle

7.1 Simple Pidgeonhole Principle

The simple form of the pidgeonhole principle is fairly obvious:

Theorem 24 (Pidgeonhole Principle)**.** *If $n+1$ objects are distributed into n boxes, then at least one box contains two or more of the objects.*

There are some other obvious statements that are useful stating formally:

Lemma 25. *If n objects are put into n boxes and no box is empty, then each box contains exactly one object.*

Lemma 26. *If n objects are put into n boxes and no box gets more than one object, then each box has an object in it.*

7.2 Strong Pidgeonhole Principle

The following theorem illustrates a special case of the pidgeonhole principle:

Theorem 27. *Let $q_1, q_2, \dots, q_n \in \mathbb{Z}_{>0}$. If $q_1 + q_2 + \cdots + q_n - n + 1$ objects are distributed into n boxes, then at least one box i has at least q_i objects.*

Corollary 27.1. *Let n and r be positive integers. If $n(r-1)+1$ objects are distributed into n boxes, then at least one of the boxs contains r or more of the objects.*

This corollary can also be expressed as an averaging principle:

Corollary 27.2. *If the average of $n \in \mathbb{Z}_{\geq 0}$ integers $m_1, m_2, \dots, m_n$ is greater than $r-1$, that is: $\frac{m_1+m_2+\cdots+m_n}{n} > r-1$, then $\exists m_k : m_k \geq r$.*

7.3 Ramsey's Theorem

Ramsey's theorem is a generalization of the pidgeonhole principle named after English Logician Frank Ramsey. The most popular and simplest form of Ramsey's theorem is seen in the following:

> Of six or more people, either there are three, each pair of whom are acquainted, or there are three, each pair of whom are unacquainted.

Before we state the generalized form of the theorem, we must become familiar with the notation used.

$$K_p \to K_n, K_m$$

is read as "K_p arrows K_n, K_m". These are all complete graphs on p, n, and m vertices. For Ramsey's theorem, we will we will color each edge within the complete graph: red if vertices are acquainted and blue if vertices are unacquainted. $K_p \to K_n, K_m$ is the assertion that no matter how the edges of K_p are colored, there is always a red K_n or a blue K_m. The generalized form of Ramsey's Theorem is:

Theorem 28 (Ramsey's Theorem)**.** *If $m \geq 2$ and $n \geq 2$ are integers, then there is a positive integer p such that $K_p \to K_n, K_m$.*

This number p is referred to as a **Ramsey number** and is notated as $r(n,m)$. Note that since red and blue can be swapped, $r(n,m) = r(m,n)$. Also note that $r(2,m) = m$ and $r(n,2) = n$. These are known as **trivial Ramsey numbers**. Note that Ramsey's Theorem is simply an existence theorem, and though we know that a Ramsey number exists for every complete graph, actually computing them is non-trivial and is still an open question in mathematics.