MTH 309: Discrete Mathematics
Course Notes

Fall 2022
Drew Armstrong

Discrete Mathematics is not a very precise term; it just means *Not Calculus*. When Calculus was first developed in the mid-1600s it unleashed a revolution in applied mathematics. Culturally, however, one could say that Calculus became too successful, to the point that it crowded other ideas out of the curriculum.

In the mid-1900s there was backlash when people realized that Calculus is not very helpful for some modern problems, including the design and programming of electronic computers. At this point there was a concerted effort to bring discrete methods into the undergraduate curriculum. But how should one organize this material? In the United States it seems that we have settled on three courses:

- linear algebra,
- probability and statistics,
- discrete mathematics.

The third course is the kind that you are taking right now. I interpret the scope of this course as: "discrete methods that are particularly useful for computer science, but do not fit within linear algebra or probability and statistics". Here are the broad ideas that we will cover:

- induction and recurrence,
- logic and boolean algebra,
- arithmetic on a computer,
- methods of counting,
- graphs, networks, trees.

So that's what I mean by *Discrete Mathematics*.

Contents

1 Induction and Recursion

1.1 Steiner's Problem

As motivation for the basic ideas of induction and recursion, we will start by considering the following geometrical problems posed by Jakob Steiner in 1826.[1]

[1] *Einige Gesetze über die Theilung der Ebene und des Raumes*, Journal für die reine und angewandte Mathematik (1826), Vol 1, page 349–364.

Steiner's Problem (1826)

(1) Find the maximum number of regions that can be formed by n lines in the plane.

(2) Find the maximum number of regions that can be formed by n planes in space.

Equivalently, we can write these as follows:

(1) How many pieces can be obtained from n cuts of a round pizza?

(2) How many pieces can be obtained from n cuts of a spherical cheese?

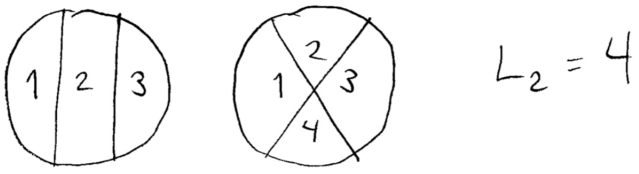
We will begin with the first problem and come back to the second problem later. How does one begin to solve a problem like this? First, we give the solution a name:

$$L_n := \max \#\text{ regions formed by } n \text{ lines in the plane.}$$

Mathematics has borrowed the symbol ":=" from the Pascal programming language. It means "is defined by to be". Now that the solution has a name, our goal is to **solve for** L_n. This could mean several things:

- give an algorithm to compute L_n,
- give a fast algorithm to compute L_n,
- give a formula for L_n,
- give a nice formula for L_n,
- etc.

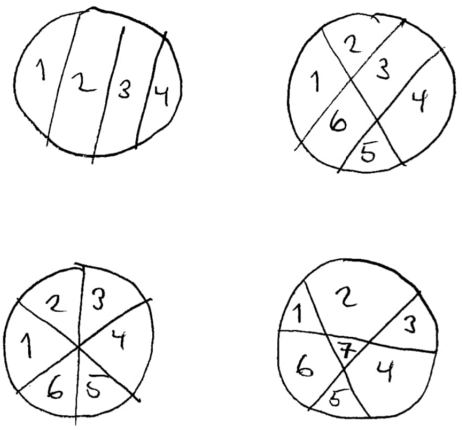
To analyze any discrete problem we always begin with **experiments**. Obviously we have $L_0 = 1$ and $L_1 = 2$. The following diagram shows that $L_2 = 4$:



At this point we might make the following guess ("conjecture"):

$$\boxed{L_n = 2^n\,?}$$

We test the conjecture by continuing the experiment:



It seems that $L_3 = 7 \neq 2^3$. So our guess was **wrong**. After some more experiments we obtain the following table of observations:

n	0	1	2	3	4	5	6
L_n	1	2	4	7	11	16	?

It is difficult to guess a formula for this, but we do observe the following pattern:

$$\boxed{L_n = L_{n-1} + n \quad \text{for all } n \geqslant 1.}$$

Now there are two issues:

- **Why** is this true? (Maybe you already have an idea.)
- Can we use it to **solve for** L_n**?**

We'll deal with the second issue first because it's more interesting. The sequence of numbers $L_0, L_1, L_2, \ldots$ is completely determined by the following **initial condition** and **recurrence relation**:

$$L_n = \begin{cases} 1 & \text{if } n = 0, \\ L_n = L_{n-1} + n & \text{if } n \geqslant 1. \end{cases}$$

Let's see what happens if we expand the recurrence:

$$\begin{aligned} L_0 &= 1, \\ L_1 &= L_0 + 1 = 1 + 1, \\ L_2 &= L_1 + 2 = (1 + 1) + 2, \\ L_3 &= L_2 + 3 = (1 + 1 + 2) + 3, \\ L_4 &= L_3 + 4 = (1 + 1 + 2 + 3) + 4, \end{aligned}$$

$$
\begin{aligned}
&\vdots \\
L_n &= L_{n-1} + n \\
&= (1 + 1 + 2 + 3 + \cdots + (n-1)) + n \\
&= 1 + (1 + 2 + 3 + \cdots + n) \\
&= 1 + \sum_{k=1}^{n} k.
\end{aligned}
$$

Yay, we have obtained a formula for L_n:

$$\boxed{L_n = 1 + \sum_{k=1}^{n} k.}$$

But is this a "good formula"? Right now it seems like there is no better way to compute L_n than to add up the numbers $1 + 2 + 3 + \cdots + n$ and then add 1 to the result. That could take a while. Luckily, there is a shortcut.

Sum of Consecutive Integers

For all integers $n \geqslant 1$ we have

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

We can prove this with a clever trick.

Proof. Let $S(n) = 1 + 2 + \cdots + n$ denote the sum of the first n integers. Now consider the quantity $2 \cdot S(n)$:

$$2 \cdot S(n) = S(n) + S(n) = \begin{array}{cccc} (1 & +2 & +\cdots & +n) \\ +(n & +(n-1) & +\cdots & +1). \end{array}$$

By arranging the terms vertically instead of horizontally we obtain

$$
\begin{aligned}
2 \cdot S(n) &= \begin{pmatrix} 1 \\ +n \end{pmatrix} + \begin{pmatrix} 2 \\ +(n-1) \end{pmatrix} + \cdots + \begin{pmatrix} n \\ +1 \end{pmatrix} \\
&= \underbrace{(n+1) + (n+1) + \cdots + (n+1)}_{n \text{ times}} \\
&= n \cdot (n+1),
\end{aligned}
$$

and hence

$$2 \cdot S(n) = n(n+1)$$
$$S(n) = n(n+1)/2.$$

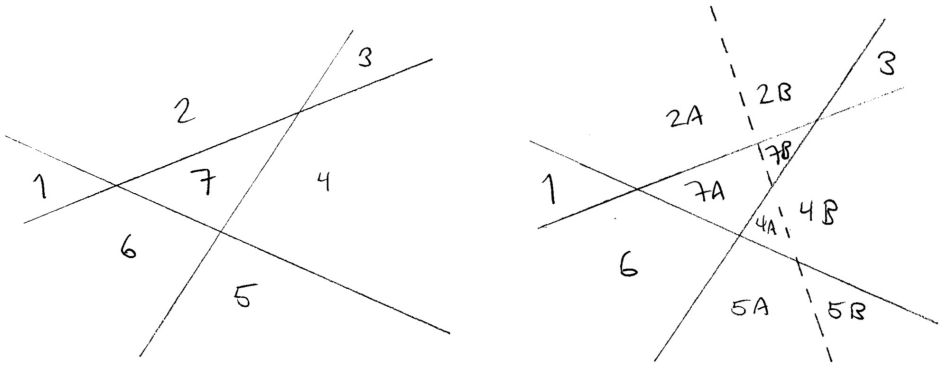
□

Then by applying this theorem we obtain a "closed formula" for the maximum number of regions formed by n lines in the plane:[2]

$$\boxed{L_n = 1 + \sum_{k=1}^{n} k = 1 + \frac{n(n+1)}{2} = \frac{n^2+n+2}{2}.}$$

I call this a "closed formula" because it involves a fixed number of operations. There is no summation or product symbol, and no "dot dot dot".

Finally, let us return to the issue of **why** the recurrence $L_n = L_{n-1} + n$ is correct. The idea of the proof is easy to explain: Suppose we already have n lines dividing the plane into L_n regions. After deleting one of the lines we will have $n-1$ lines cutting the plane into L_{n-1} regions. By putting the nth line back we will obtain one new region for each of the L_{n-1} regions that the nth line intersects. For example, see the following diagram:



On the left we have 3 lines and $L_3 = 7$ regions. On the right we see that the 4-th line cuts through 4 regions, adding 4 more to the total: $L_4 = L_3 + 4 = 7 + 4 = 11$. In general, when we add the nth line it will cut through all $n-1$ of the previous lines, and hence it will cut through n of the previous regions, adding n regions to the total: $L_n = L_{n-1} + n$. I think it would be too much work to turn this into a formal proof, so let's not bother.

Instead, let's return to Steiner's second problem. Define the following notation:[3]

$$K_n := \max \#\text{ regions formed by } n \text{ planes in space.}$$

[2]Remark: It follows from this formula that $n^2 + n + 2$ is always an even number. Do you see why?

[3]The notation L_n was for "lines". The notation K_n is for "Kugel" (ball/sphere) or "Käse" (cheese).

It's much harder to draw 3D pictures, but you can probably convince yourself of the following:

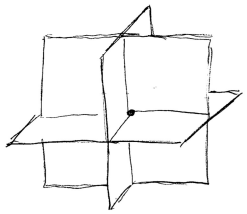
n	0	1	2	3	4
K_n	1	2	4	8	?

You might again conjecture the following pattern:

$$\boxed{K_n = 2^n\,?}$$

But again you would be **wrong**. Try as you might, you will not be able to get 16 pieces of cheese from 4 cuts. It turns out that the maximum is $K_4 = 15$. How do I know this?

Suppose that you have 3 planes dividing space into 8 regions, as follows:



If we add a 4th plane then we will obtain one new region for each old region that the new plane passes through. But it is impossible to cut through all 8 regions at the same time; the best we can do is 7. Indeed, the 3 original planes will form 3 lines in the 4th plane, and we already know the best solution to this problem is $L_3 = 7$. In general, I claim that we have the following recurrence:

$$K_n = \begin{cases} 1 & \text{if } n = 0, \\ K_{n-1} + L_{n-1} & \text{if } n \geqslant 1. \end{cases}$$

It's not so important that you believe me. Let's just use this recurrence to **solve for** K_n. We start by expanding the first few terms until we see a pattern:

$$\begin{aligned}
K_0 &= 1, \\
K_1 &= 1 + L_0, \\
K_2 &= K_1 + L_1 = (1 + L_0) + L_1, \\
K_3 &= K_2 + L_2 = (1 + L_0 + L_1) + L_2, \\
K_4 &= K_3 + L_3 = (1 + L_0 + L_1 + L_2) + L_3, \\
&\;\;\vdots \\
K_n &= K_{n-1} + L_{n-1} \\
&= (1 + L_0 + L_2 + L_3 + \cdots + L_{n-2}) + L_{n-1} \\
&= 1 + (L_0 + L_1 + L_2 + \cdots + L_{n-1}) \\
&= 1 + \sum_{k=0}^{n-1} L_k.
\end{aligned}$$

Then we apply our previous formula $L_k = (k^2 + k + 2)/2$ to obtain

$$\begin{aligned} K_n &= 1 + \sum_{k=0}^{n-1} \frac{k^2 + k + 2}{2} \\ &= 1 + \frac{1}{2}\left(\sum_{k=0}^{n-1} k^2 + \sum_{k=0}^{n-1} k + \sum_{k=0}^{n-1} 2 \right). \end{aligned}$$

Two of these sums can be simplified. We know that

$$\sum_{k=0}^{n-1} 2 = \underbrace{2 + 2 + \cdots + 2}_{n \text{ times}} = 2n$$

and

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

Thus we obtain

$$\begin{aligned} K_n &= 1 + \frac{1}{2}\left(\sum_{k=0}^{n-1} k^2 + \sum_{k=0}^{n-1} k + \sum_{k=0}^{n-1} 2 \right) \\ &= 1 + \frac{1}{2}\left(\sum_{k=0}^{n-1} k^2 + \frac{n(n-1)}{2} + 2n \right) \\ &= \frac{1}{2} \sum_{k=0}^{n-1} k^2 + \frac{n(n-1)}{4} + n + 1. \end{aligned}$$

But now we are stuck. In order to go further we need a closed formula for the sum of consecutive squares:

$$\sum_{k=0}^{n-1} = 0^2 + 1^2 + 2^2 + \cdots + (n-1)^2 = 1^2 + 2^2 + \cdots + (n-1)^2 = ?$$

I will tell you the answer next time.

1.2 The Principle of Induction

I have learned from experience that most students do not understand the principle of induction on the first try. For this reason I like to discuss it early and often.

Principle of Induction

Let $P(n)$ be a sequence of mathematical statements[4] and suppose that the following two properties hold:

- *Base Case.* $P(b)$ is a true statement for some specific b.

- *Induction Step.* If $n \geqslant b$ then the truth of $P(n)$ implies the truth of $P(n+1)$.

Then we conclude that $P(n)$ is true **for all** $n \geqslant b$.

Here is the most famous example. Consider the following sequence of statements:

$$P(n) := \text{“}1 + 2 + \cdots + n = \frac{n(n+1)}{2}\text{”}.$$

Last time I showed you a clever trick to prove that $P(n)$ is true for all $n \geqslant 1$. But what if we can't find a clever trick? Then we can still prove the statement “by induction”.

Proof by Induction. The base case is $b = 1$. Note that $P(b)$ is a true statement because

$$P(b) = P(1) = \text{“}1 = \frac{1 \cdot 2}{2}\text{”} = \text{“}1 = 1\text{”}.$$

Now we fix some integer $n \geqslant 1$ and **assume for induction** that $P(n)$ is a true statement. In other words, we assume that

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

In this (hypothetical) case, we want to prove that $P(n+1)$ is also true. In other words, we want to prove that the following equation is true:

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

How? In this example we have very few ingredients to work with. After some trial and error you will discover the following argument:

$$\begin{aligned} 1 + 2 + \cdots + (n+1) &= 1 + 2 + \cdots + n + (n+1) \\ &= (1 + 2 + \cdots + n) + (n+1) \\ &= \frac{n(n+1)}{2} + (n+1) \\ &= (n+1)\left(\frac{n}{2} + 1\right) \\ &= \frac{(n+1)(n+2)}{2}. \end{aligned}$$

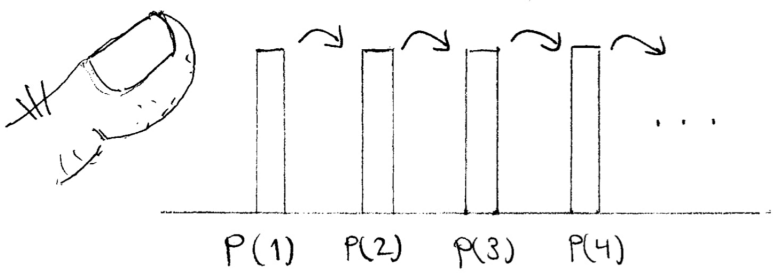
In other words, $P(n+1)$ is true. Since we have verified that the sequence $P(n)$ satisfies both properties (the base case and the induction step), we conclude that the statement $P(n)$ is true for all $n \geqslant b = 1$. □

What did we just do? Here is an informal discussion:

[4] A mathematical statement is a sentence that is either “true” or “false”. Some sentences (for example: “the weather is nice today”) are not mathematical statements. We will discuss this in the next chapter.

- After some experiments we might guess that the statement $P(n)$ = "$1 + 2 + \cdots + n = n(n+1)/2$" is true for all positive whole numbers n.
- We check by hand that the statements $P(1), P(2), \ldots, P(10)$ are all true.
- Just to be sure, we have our computer check that the statements $P(1)$, $P(2)$, ... $P(1000000)$ are all true. But this is still not a proof. Maybe the statement $P(100001)$ is false?
- So we set up our computer to keep checking every statement $P(1), P(2), P(3), \ldots$ and never stop. Eventually the computer will melt. Suppose that $P(n)$ is the last statement the computer checked before it melted.
- This n is a big number, but we still don't know for sure if $P(n+1)$ is a true.
- Now the induction step takes over. Even if we don't know the exact value of n we can write an abstract proof to show that $P(n)$ implies $P(n+1)$. Since the argument works for any value of n we conclude that the statements are true forever.
- Summary: The base case is a computer that verifies enough statements to get us started. But all the computers in the world can only check a finite number of cases. The induction step is an abstract argument that takes us from there to infinity.

If you didn't like that, here is a picture:



- The sequence of statements is a row of **dominoes**.
- The base case is **your finger**, which knocks down at least one domino.
- The induction step is **gravity**, which guarantees that the dominoes keep falling forever.

To practice the concept, you should use induction to prove the following theorem. The proof is "exactly the same" as the proof for the sum of consecutive integers. Next time I'll tell you where this formula comes from.

Sum of Consecutive Squares

For all integers $n \geqslant 1$ we have

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

Finally, we obtain a closed formula for the maximum number of regions formed by n planes in space, or the maximum number of pieces that can be made from n cuts of a spherical cheese:

$$\begin{aligned}
K_n &= \frac{1}{2}\left[\sum_{k=0}^{n-1} k^2\right] + \frac{n(n-1)}{4} + (n+1) \\
&= \frac{1}{2}\left[\frac{(n-1)((n-1)+1)(2(n-1)+1)}{6}\right] + \frac{n(n-1)}{4} + (n+1) \\
&= \frac{1}{2}\cdot\frac{n(n-1)(2n-1)}{6} + \frac{n(n-1)}{4} + (n+1) \\
&= (n-1)\left[\frac{1}{2}\cdot\frac{n(2n-1)}{6} + \frac{n}{4}\right] + (n+1) \\
&= (n-1)\left[\frac{2n^2-n+3n}{12}\right] + (n+1) \\
&= (n-1)\left[\frac{2n(n+1)}{12}\right] + (n+1) \\
&= \frac{(n-1)n(n+1)}{6} + (n+1) \\
&= (n+1)\left[\frac{n(n-1)}{6} + 1\right] \\
&= \frac{(n+1)(n^2-n+6)}{6}.
\end{aligned}$$

1.3 Sums of Powers

Last time I showed you a mysterious formula for the sum of consecutive squares, and you proved it by induction. Today I'll show you where the formula comes from. It turns out that naming things is the key to the problem.

Sum of Consecutive p-th Powers

We will use the following notation for the sum of the first n consecutive p-th powers:

$$S_p(n) := \sum_{k=1}^{n} k^p = 1^p + 2^p + 3^p + \cdots + n^p.$$

We could also define these numbers by the following initial condition and recurrence:

$$S_p(n) := \begin{cases} 1 & \text{if } n = 1, \\ S_p(n-1) + n^p & \text{if } n \geqslant 2. \end{cases}$$

In order to practice this notation, let me show you yet another proof of the identity

$$S_1(n) = \frac{n(n+1)}{2}.$$

Proof. The trick is to expand the sum $S_2(n+1)$ in two different ways. On the one hand, we have the basic recurrence:

$$S_2(n+1) = 1^2 + 2^2 + \cdots + (n+1)^2 = (1^2 + 2^2 + \cdots n^2) + (n+1)^2 = S_2(n) + (n+1)^2.$$

On the other hand, we can change the index of summation as follows:

$$\begin{aligned} S_2(n+1) &= \sum_{k=1}^{n+1} k^2 \\ &= \sum_{\ell=0}^{n} (\ell+1)^2 && \ell := k-1 \\ &= \sum_{\ell=0}^{n} (\ell^2 + 2\ell + 1) \\ &= \sum_{\ell=0}^{n} \ell^2 + 2 \cdot \sum_{\ell=0}^{n} \ell + \sum_{\ell=0}^{n} 1. \end{aligned}$$

In the first two sums we can change the lower limit from $\ell = 0$ to $\ell = 1$ because the $\ell = 0$ terms are zero. Then we have names for all three terms:

$$\begin{aligned} S_2(n+1) &= \sum_{\ell=0}^{n} \ell^2 + 2 \cdot \sum_{\ell=0}^{n} \ell + \sum_{\ell=0}^{n} 1 \\ &= \sum_{\ell=1}^{n} \ell^2 + 2 \cdot \sum_{\ell=1}^{n} \ell + \sum_{\ell=0}^{n} 1 \\ &= S_2(n) + 2 \cdot S_1(n) + (n+1). \end{aligned}$$

Finally, we equate the two expressions for $S_2(n+1)$ to obtain

$$\begin{aligned}
S_2(n) + 2 \cdot S_1(n) + (n+1) &= S_2(n) + (n+1)^2 \\
\cancel{S_2(n)} + 2 \cdot S_1(n) + (n+1) &= \cancel{S_2(n)} + (n+1)^2 \\
2 \cdot S_1(n) + (n+1) &= (n+1)^2 \\
2 \cdot S_1(n) &= (n+1)^2 - (n+1) \\
2 \cdot S_1(n) &= ((n+1)-1)(n+1) \\
2 \cdot S_1(n) &= n(n+1) \\
S_1(n) &= n(n+1)/2.
\end{aligned}$$

It was lucky that the unknown term $S_2(n)$ appeared on both sides. □

Now let's use "exactly the same trick" to prove the formula

$$S_2(n) = \frac{n(n+1)(2n+1)}{6}.$$

Proof. We will expand $S_3(n+1)$ in two different ways. On the one hand, we have

$$S_3(n+1) = S_3(n) + (n+1)^3.$$

On the other hand, we have

$$\begin{aligned}
S_3(n+1) &= \sum_{k=1}^{n+1} k^3 \\
&= \sum_{\ell=0}^{n} (\ell+1)^3 && \ell := k-1 \\
&= \sum_{\ell=0}^{n} (\ell^3 + 3\ell^2 + 3\ell + 1) \\
&= \sum_{\ell=0}^{n} \ell^3 + 3 \cdot \sum_{\ell=0}^{n} \ell^2 + 3 \cdot \sum_{\ell=0}^{n} \ell + \sum_{\ell=0}^{n} 1 \\
&= \sum_{\ell=1}^{n} \ell^3 + 3 \cdot \sum_{\ell=1}^{n} \ell^2 + 3 \cdot \sum_{\ell=1}^{n} \ell + \sum_{\ell=0}^{n} 1 \\
&= S_3(n) + 3 \cdot S_2(n) + 3 \cdot S_1(n) + (n+1).
\end{aligned}$$

In the third line we used the algebraic identity $(\ell+1)^3 = \ell^3 + 3\ell^2 + 3\ell + 1$. I will show you a quick way to compute this next time. Finally, we equate the two expressions for $S_3(n+1)$ and plug in the known value of $S_1(n)$ to obtain

$$S_3(n) + 3 \cdot S_2(n) + 3 \cdot S_1(n) + (n+1) = S_3(n) + (n+1)^3$$

$$
\begin{aligned}
\cancel{S_3(n)} + 3 \cdot S_2(n) + 3 \cdot S_1(n) + (n+1) &= \cancel{S_3(n)} + (n+1)^3 \\
3 \cdot S_2(n) + 3 \cdot S_1(n) + (n+1) &= (n+1)^3 \\
3 \cdot S_2(n) &= (n+1)^3 - (n+1) - 3 \cdot S_1(n) \\
3 \cdot S_2(n) &= (n+1)^3 - (n+1) - 3 \cdot \frac{n(n+1)}{2} \\
3 \cdot S_2(n) &= \frac{(n+1)}{2}\left[2(n+1)^2 - 2 - 3n\right] \\
3 \cdot S_2(n) &= \frac{(n+1)}{2}\left[2(n^2+2n+1) - 2 - 3n\right] \\
3 \cdot S_2(n) &= \frac{(n+1)}{2}\left[2n^2 + n\right] \\
3 \cdot S_2(n) &= \frac{n(n+1)(2n+1)}{2} \\
S_2(n) &= \frac{n(n+1)(2n+1)}{6}.
\end{aligned}
$$

It was lucky that the unknown term $S_3(n)$ appeared on both sides. ▫

In the previous section we saw that this formula helps to solve Steiner's 1826 problem on the maximum number of regions formed by n planes in space. But the original motivation to consider sums of p-th powers comes from the early history of Calculus. In the year 1636, Pierre de Fermat stated the following result in a letter to Gilles de Roberval.

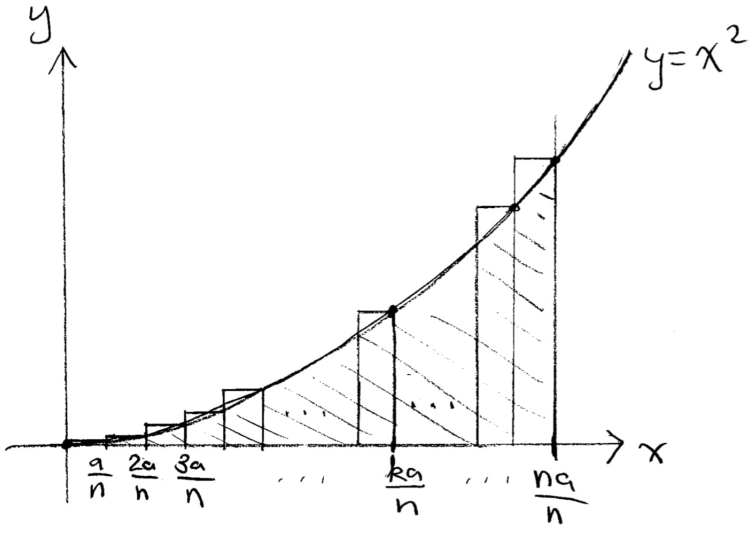
Fermat's Theorem (1636)

Let p be a positive integer. The area under the graph of the "higher parabola" $y = x^p$ from $x = 0$ to $x = a$ is equal to $a^{p+1}/(p+1)$. In modern terminology we would say that

$$\int x^p \, dx = \frac{x^{p+1}}{p+1}.$$

This was significant because it was the first major progress on the theory of integrals since the classical work of Archimedes. To illustrate Fermat's method we will consider the case of the ordinary parabola[5] $y = x^2$. In order to compute the area under $y = x^2$ we divide the interval between $x = 0$ and $x = a$ into n equal pieces. Then on each piece we draw a rectangle of width a/n with height roughly equal to the height of the graph:

[5]This case **was** known to Archimedes, but the method is new.



Note that the k-th rectangle has height $(ka/n)^2$ and hence

$$(\text{area of } k\text{-th rectangle}) = (\text{base})(\text{height}) = (a/n)(ka/n)^2 = \frac{a^3}{n^3} \cdot k^2.$$

Using our formula for $S_2(n)$ gives the total area of all n rectangles:

$$\begin{aligned}(\text{total area of rectangles}) &= \sum_{k=1}^{n} (\text{area of } k\text{-th rectangle}) \\ &= \sum_{k=1}^{n} \frac{a^3}{n^3} \cdot k^2 \\ &= \frac{a^3}{n^3} \cdot \sum_{k=1}^{n} k^2 \\ &= \frac{a^3}{n^3} \cdot \frac{n(n+1)(2n+1)}{6}.\end{aligned}$$

Note that the area of the rectangles is approximately equal to the area under the curve. Furthermore, the approximation is more accurate for large values of n. In order to see what happens as n grows, it is more convenient to rewrite our formula for $S_2(n)$ as follows:

$$S_2(n) = \frac{n(n+1)(2n+1)}{6} = \frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n.$$

Then we have

$$(\text{total area of rectangles}) = \frac{a^3}{n^3} \cdot \left(\frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n\right) = \frac{a^3}{3} + \boxed{\frac{1}{2n} + \frac{1}{6n^2}.}$$

Note that the quantity in the box goes to zero as n goes to infinity. Fermat concluded that in the limit we must have

$$\begin{array}{ccc} \text{(total area of rectangles)} & \to & \text{(area under the curve)}, \\ \frac{a^3}{3} + \frac{1}{2n} + \frac{1}{6n^2} & \to & \frac{a^3}{3} + 0. \end{array}$$

Actually, Fermat did not need a closed form for $S_2(n)$; he only needed to know that

$$S_2(n) = \frac{n^3}{3} + \text{lower terms}.$$

More generally, Fermat's theorem is equivalent to the fact that

$$S_p(n) = \frac{n^{p+1}}{p+1} + \text{lower terms}.$$

In the meantime, curious mathematicians such as Johann Faulhaber (*Academia Algebræ*, 1631) were trying to find a **closed formula** for the sum of p-th powers. This can be accomplished with "exactly the same trick" that we used above, i.e., by expanding the sum $S_{p+1}(n+1)$ in two different ways and then by solving for the unknown $S_p(n)$. The resulting formula is not pretty and it also contains two pieces of undefined notation, but I will include it here for the curious. It was first written down in this form by Jakob Bernoulli (published posthumously in the *Ars Conjectandi*, 1731).

Bernoulli's Formula (1731)

For all positive integers n and p we have

$$S_p(n) = 1^p + 2^p + \cdots + n^p = \frac{n^{p+1}}{p+1} + \boxed{\sum_{k=1}^{p-1} \binom{p+1}{k} \cdot B_k \cdot n^{p+1-k}}.$$

The expression in the box are the "lower terms" that Fermat did not need to compute.

The first undefined piece of notation is the sequence of *Bernoulli numbers*, which were invented by Bernoulli for exactly this purpose:

n	0	1	2	3	4	5	6	7	8	9	10
B_n	1	$\frac{1}{2}$	$\frac{1}{6}$	0	$\frac{-1}{30}$	0	$\frac{1}{42}$	0	$\frac{-1}{30}$	0	$\frac{5}{66}$

Don't be discouraged if you can't see a pattern here. The recurrence is very complicated:

$$B_n := \begin{cases} 1 & \text{if } n = 0, \\ 1 - \sum_{k=0}^{n-1} \frac{1}{n-k+1} \cdot \binom{n}{k} \cdot B_k & \text{if } n \geqslant 1. \end{cases}$$

The second undefined piece of terminology are the numbers $\binom{n}{k}$, which occur in Bernoulli's formula and in the recurrence for the Bernoulli numbers. These numbers are the subject of the next section.

1.4 Pascal's Triangle

The numbers $\binom{n}{k}$ appearing in Bernoulli's formula have many different interpretations. The easiest way to define them is by recurrence.

Definition of Pascal's Triangle

For all integers $0 \leqslant k \leqslant n$ we define the number $\binom{n}{k}$ by the following **boundary conditions** and **recurrence relation**:

$$\binom{n}{k} := \begin{cases} 1 & \text{if } k = 0 \text{ or } k = n, \\ \binom{n-1}{k-1} + \binom{n-1}{k} & \text{otherwise.} \end{cases}$$

We think of these numbers as forming an infinite triangular array with 1s down the sides, where each number inside the triangle is defined as the sum of the two numbers above it:

$$\begin{array}{ccccccccc} & & & \binom{0}{0} & & & & & \\ & & \binom{1}{0} & & \binom{1}{1} & & & & \\ & \binom{2}{0} & & \binom{2}{1} & & \binom{2}{2} & & & \\ \binom{3}{0} & & \binom{3}{1} & & \binom{3}{2} & & \binom{3}{3} & & \\ \binom{4}{0} & & \binom{4}{1} & & \binom{4}{2} & & \binom{4}{3} & & \binom{4}{4} \end{array} = \begin{array}{ccccccccc} & & & 1 & & & & & \\ & & 1 & & 1 & & & & \\ & 1 & & 2 & & 1 & & & \\ 1 & & 3 & & 3 & & 1 & & \\ 1 & & 4 & & 6 & & 4 & & 1 \end{array}$$

These numbers have been known in many times and places. The name *Pascal's triangle* comes from French mathematician Blaise Pascal, who wrote a systematic work on these numbers in 1654 (*Treatise on the arithmetical triangle*).[6] Pascal's primary motivation was the study of probability and games of chance, but he also included a section on the sums of p-th powers.

We will have much more to say about Pascal's triangle in Chapter 4. For now let me give you two algebraic interpretations of the numbers $\binom{n}{k}$, both of which can be proved by induction. The first interpretation explains why we call these numbers *binomial coefficients*.

[6] The symbol $\binom{n}{k}$ for these numbers was introduced much later by von Ettingshausen, *Die combinatorische Analysis als Vorbereitungslehre zum Studium der theoretischen höhern Mathematik* (1826), page 30.

The Binomial Theorem

Let x be any number and observe the following expansions:

$$\begin{aligned}(1+x)^0 &= 1,\\ (1+x)^1 &= 1+x,\\ (1+x)^2 &= 1+2x+x^2,\\ (1+x)^3 &= 1+3x+3x^2+x^3,\\ (1+x)^4 &= 1+4x+6x^2+4x^3+x^4.\end{aligned}$$

After a while you may notice that the coefficients in these expansions are the same as the entries of Pascal's Triangle. To be specific, the coefficient of x^k in the expansion of $(1+x)^n$ is equal to the k-th entry in the n-th row of Pascal's Triangle:

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

Proof. Let $0 \leqslant k \leqslant n$ be whole numbers and let $c(n,k)$ be the coefficient of x^k in the expansion of $(1+x)^n$. In order to prove that $c(n,k) = \binom{n}{k}$ it is enough to show that these numbers satisfy the same boundary conditions and recurrence.

- *Boundary Conditions.* Note that we always have

  $$(1+x)^n = 1x^0 + \text{middle terms} + 1x^n.$$

  In other words we have $c(n,0) = c(n,n) = 1$.

- *Recurrence Relation.* Now let $0 < k < n$ and consider the following algebraic identity:

  $$\begin{aligned}(1+x)^n &= (1+x)(1+x)^{n-1}\\ (1+x)^n &= x(1+x)^{n-1} + (1+x)^{n-1}.\end{aligned}$$

  Now let's compute the coefficient of x^k on each side. By definition we know that $c(n,k)$ is the coefficient of x^k in $(1+x)^n$. By definition we also know that $c(n-1,k)$ is the coefficient of x^k in $(1+x)^{n-1}$. And what about $x(1+x)^{n-1}$? We observe that

  $$\begin{aligned}x(1+x)^{n-1} &= x\Big[c(n-1,0)x^0 + \cdots + c(n-1,k-1)x^{k-1} + \cdots + c(n-1,n-1)x^{n-1}\Big]\\ x(1+x)^{n-1} &= c(n-1,0)x^1 + \cdots + c(n-1,k-1)x^k + \cdots + c(n-1,n-1)x^n,\end{aligned}$$

  hence the coefficient of x^k in $x(1+x)^{n-1}$ is $c(n-1,k-1)$. In summary, we have

  $$\begin{aligned}&(\text{coefficient of } x^k \text{ in } (1+x)^n) = (\text{coefficient of } x^k \text{ in } x(1+x)^{n-1} + (1+x)^{n-1}),\\ &c(n,k) = c(n-1,k-1) + c(n-1,k).\end{aligned}$$

□

The next algebraic interpretation gives a "closed formula" for $\binom{n}{k}$ in terms of "factorials".

Closed Formula for Binomial Coefficients

For all integers $n \geqslant 0$ we define the *factorial* by the following recurrence:

$$n! := \begin{cases} 1 & \text{if } n = 0, \\ n \cdot (n-1)! & \text{if } n \geqslant 1. \end{cases}$$

In other words, we have $0! = 1$ and

$$n! = n(n-1)(n-2)\cdots 3 \cdot 2 \cdot 1 \qquad \text{for } n \geqslant 1.$$

I claim that the binomial coefficients satisfy the following formula:

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

Proof. Let $0 \leqslant k \leqslant n$. Again, it suffices to show that this formula satisfies the same boundary conditions and recurrence:

- *Boundary Conditions.* If $k = 0$ or $n = 0$ then since $0! = 1$ we have

$$\frac{n!}{0!(n-0)!} = \frac{n!}{0!n!} = \frac{n!}{n!} = 1 \qquad \text{and} \qquad \frac{n!}{n!(n-n)!} = \frac{n!}{n!0!} = \frac{n!}{n!} = 1.$$

- *Recurrence Relation.* Now let $0 < k < n$. In the following calculation we use the identities $k! = k(k-1)!$ and $(n-k)! = (n-k)(n-k-1)!$ to find a common denominator for the sum of fractions:

$$\begin{aligned} &\frac{(n-1)!}{(k-1)!((n-1)-(k-1))!} + \frac{(n-1)!}{k!(n-1-k)!} \\ &= \frac{(n-1)!}{(k-1)!(n-k)!} + \frac{(n-1)!}{k!(n-k-1)!} \\ &= \frac{k}{k} \cdot \frac{(n-1)!}{(k-1)!(n-k)!} + \frac{(n-1)!}{k!(n-k-1)!} \cdot \frac{n-k}{n-k} \\ &= k \cdot \frac{(n-1)!}{k!(n-k)!} + \frac{(n-1)!}{k!(n-k)!} \cdot (n-k) \\ &= \frac{k(n-1)! + (n-k)(n-1)!}{k!(n-k)!} \end{aligned}$$

$$
\begin{aligned}
&= \frac{(\cancel{k} + n - \cancel{k})(n-1)!}{k!(n-k)!} \\
&= \frac{n(n-1)!}{k!(n-k)!} \\
&= \frac{n!}{k!(n-k)!}.
\end{aligned}
$$

□

The binomial coefficients are extremely useful for simplifying complicated formulas. For a striking example of this, let's return to Steiner's problem. Recall that we have proved the following facts:

$$
\begin{aligned}
L_n &= (\text{max \# regions formed by } n \text{ lines in the plane}) = \frac{n^2+n+2}{2}, \\
K_n &= (\text{max \# regions formed by } n \text{ planes in space}) = \frac{(n+1)(n^2-n+6)}{6}.
\end{aligned}
$$

Do you see a pattern? Actually, Steiner expressed these formulas in a slightly different way:

$$
\begin{aligned}
L_n &= \frac{n(n-1)}{2 \cdot 1} + n + 1, \\
K_n &= \frac{n(n-1)(n-2)}{3 \cdot 2 \cdot 1} + \frac{n(n-1)}{2 \cdot 1} + n + 1.
\end{aligned}
$$

We can make this even cleaner with the following trick.

Alternative Formula for Binomial Coefficients

By canceling the factor $(n-k)!$ from the numerator and denominator, we obtain

$$
\begin{aligned}
\binom{n}{k} &= \frac{n!}{k!(n-k)!} \\
&= \frac{n(n-1)(n-2)\cdots(n-k+1)\cancel{(n-k)(n-k-1)\cdots 3 \cdot 2 \cdot 1}}{k(k-1)\cdots 3 \cdot 2 \cdot 1 \cdot \cancel{(n-k)(n-k-1)\cdots 3 \cdot 2 \cdot 1}} \\
&= \frac{n(n-1)(n-2)\cdots(n-k+1)}{k(k-1)(k-2)\cdots 1}.
\end{aligned}
$$

Then Steiner's formulas become

$$
L_n = \binom{n}{2} + \binom{n}{1} + \binom{n}{0},
$$

$$K_n = \binom{n}{3} + \binom{n}{2} + \binom{n}{1} + \binom{n}{0}.$$

Now do you see a pattern? In the year 1826 there was no such thing as "4-dimensional space". However, in the 1840s, a younger colleague and fellow Swiss geometer of Steiner's named Ludwig Schläfli was bold enough to state Steiner's result in full generality.[7]

The Steiner-Schläfli Theorem (1850)

The maximum number of regions formed by n hyperplanes[8] in d-dimensional space is

$$\binom{n}{d} + \binom{n}{d-1} + \binom{n}{d-2} + \cdots + \binom{n}{1} + \binom{n}{0}.$$

Furthermore, the maximum number of **bounded** regions is

$$\binom{n}{d} - \binom{n}{d-1} + \binom{n}{d-2} - \cdots + (-1)^{d-1}\binom{n}{1} + (-1)^d\binom{n}{0}.$$

For the purpose of this theorem we define the symbol $\binom{n}{k}$ to be zero whenever $k > n$. You will investigate the case $d > n$ on the homework.

For example, let $d = 2$ and $n = 4$. By looking at the 4-th row of Pascal's triangle we find that 4 lines can divide the plane into

$$\binom{4}{2} + \binom{4}{1} + \binom{4}{0} = 6 + 4 + 1 = 11 \text{ regions.}$$

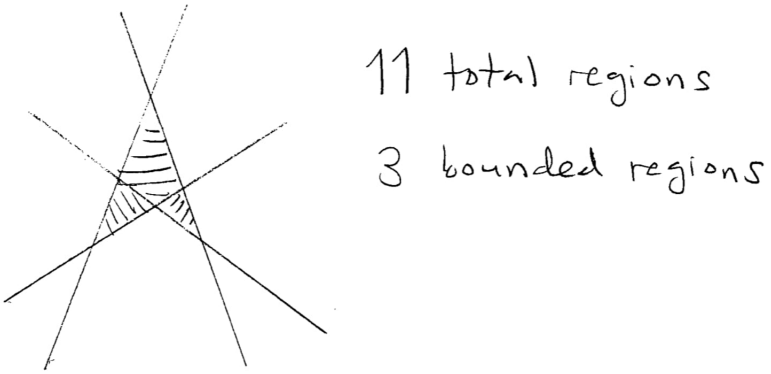
Furthermore, among these 11 regions the number that are **bounded** is

$$\binom{4}{2} - \binom{4}{1} + \binom{4}{0} = 6 - 4 + 1 = 3.$$

Here is a picture:

[7] Schläfli, *Theorie der vielfachen Kontinuität* (1850), Section 16.

[8] A *hyperplane* is a flat $(d-1)$-dimensional subset of d-dimensional space. It can be used to cut the space into two pieces.



For another example, let $d = 4$ and $n = 6$. Then Schläfli's formula tells us that 6 hyperplanes in 4-dimensional space can form at most

$$\binom{6}{4} + \binom{6}{3} + \binom{6}{2} + \binom{6}{1} + \binom{6}{0} = 15 + 20 + 15 + 6 + 1 = 57 \text{ total regions}$$

and

$$\binom{6}{4} - \binom{6}{3} + \binom{6}{2} - \binom{6}{1} + \binom{6}{0} = 15 - 20 + 15 - 6 + 1 = 5 \text{ bounded regions.}$$

You can decide for yourself whether this makes any sense.

1.5 Worked Exercises

1.1. Simplify the following sum as much as possible:

$$\sum_{k=0}^{n} \frac{(k+1)(k+2)}{2} = ?$$

Solution. Recall that we have the following formulas:

$$\sum_{k=0}^{n} k^2 = \sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}, \quad \sum_{k=0}^{n} k = \sum_{k=1}^{n} k = \frac{n(n+1)}{2}, \quad \sum_{k=0}^{n} = n+1.$$

By combining these formulas we obtain

$$\begin{aligned}\sum_{k=0}^{n} \frac{(k+1)(k+2)}{2} &= \frac{1}{2}\sum_{k=0}^{n}(k+1)(k+2) \\ &= \frac{1}{2}\sum_{k=0}^{n}(k^2+3k+2)\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{2}\left[\sum_{k=0}^{n} k^2 + 3\sum_{k=0}^{n} k + 2\sum_{k=0}^{n} 1\right] \\
&= \frac{1}{2}\left[\frac{n(n+1)(2n+1)}{6} + 3\frac{n(n+1)}{2} + 2(n+1)\right] \\
&= \frac{n+1}{2}\left[\frac{n(2n+1)}{6} + 3\frac{n}{2} + 2\right] \\
&= \frac{n+1}{2}\left[\frac{2n^2+n+9n+12}{6}\right] \\
&= \frac{n+1}{2}\left[\frac{2n^2+10n+12}{6}\right] \\
&= \frac{n+1}{6}\left[n^2+5n+6\right] \\
&= \frac{n+1}{6}(n+2)(n+3) \\
&= \frac{(n+1)(n+2)(n+3)}{6}.
\end{aligned}
$$

Just for fun, we can express this result in terms of binomial coefficients:

$$\sum_{k=0}^{n} \binom{k+2}{2} = \binom{n+3}{3}.$$

Maybe we will see a generalization of this identity later.

1.2. Find a closed formula for the sum of the first n odd numbers:

$$1 + 3 + 5 + 7 + \cdots + (2n-1) = ?$$

Solution. We have

$$
\begin{aligned}
1 + 3 + 5 + \cdots + (2n-1) &= \sum_{k=1}^{n} (2k-1) \\
&= 2\sum_{k=1}^{n} k - \sum_{k=1}^{n} 1 \\
&= 2\frac{n(n+1)}{2} - n \\
&= n(n+1) - n \\
&= n^2.
\end{aligned}
$$

You could also guess the answer by experiment and then prove it by induction. Consider the statement

$$P(n) = \text{“}1 + 3 + 5 + \cdots + (2n-1) = n^2\text{”}.$$

- *Base Case.* $P(1) =$ "$1 = 1^2$" is a true statement.
- *Induction Step.* Fix $n \geqslant 1$ and assume for induction that $P(n)$ is true. Then we have

$$\begin{aligned} 1 + 3 + 5 + \cdots + (2(n+1) - 1) &= 1 + 3 + 5 + \cdots + (2n - 1) + (2(n+1) - 1) \\ &= n^2 + (2(n+1) - 1) \\ &= n^2 + 2n + 1 \\ &= (n+1)^2, \end{aligned}$$

  and hence $P(n+1)$ is also true.

1.3. For any integers $1 \leqslant a \leqslant b$, find a closed formula for the sum of all integers between them:

$$a + (a+1) + \cdots + (b-1) + b = ?$$

Solution. This is a difference of two sums that you already know:

$$\begin{aligned} a + (a+1) + \cdots + (b-1) + b &= (1 + 2 + \cdots + b) - (1 + 2 + \cdots + (a-1)) \\ &= \frac{b(b+1)}{2} - \frac{(a-1)a}{2}. \end{aligned}$$

We can also write this as

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

Compare this to the "continuous version":

$$\int_a^b x\,dx = \frac{b^2}{2} - \frac{a^2}{2}.$$

1.4. The sequence of *factorials* $0!, 1!, 2!, \ldots$ is defined by the following initial condition and recurrence relation:

$$n! := \begin{cases} 1 & \text{if } n = 0, \\ (n-1)! \cdot n & \text{if } n \geqslant 1. \end{cases}$$

Prove by induction that we have

$$n! > 2^n \qquad \text{for all } n \geqslant 4.$$

Solution. Consider the statement $P(n) =$ "$n! > 2^n$".

- *Base Case.* We observe that $P(4) =$ "$24 > 16$" is a true statement.

- *Induction Step.* Now fix $n \geqslant 4$ and assume for induction that $P(n) =$ "$n! > 2^n$" is true. In this case we have

$$\begin{aligned}(n+1)! &= (n+1) \cdot n! && \text{by definition} \\ &> (n+1) \cdot 2^n && \text{because } P(n) \text{ is true} \\ &> 2 \cdot 2^n && \text{because } n+1 > 2 \\ &= 2^{n+1}, \end{aligned}$$

  and hence $P(n+1)$ is also true.

1.5. The *Fibonacci sequence* $F_0, F_1, F_2, \ldots$ is defined by the following initial conditions and recurrence relation:

$$F_n := \begin{cases} 0 & \text{if } n = 0, \\ 1 & \text{if } n = 1, \\ F_{n-1} + F_{n-2} & \text{if } n \geqslant 2. \end{cases}$$

Let $\varphi := (1 + \sqrt{5})/2$ be the *golden ratio*, which satisfies $\varphi^2 = \varphi + 1$ (check it if you don't believe me). Prove by induction that we have

$$\varphi^{n-2} < F_n < \varphi^{n-1} \qquad \text{for all } n \geqslant 3.$$

[Hint: Use strong induction with two base cases.]

Induction can be stated in many equivalent ways. Here is the principle of "strong induction".

Principle of Strong Induction

Let $P(n)$ be a sequence of mathematical statements and suppose that the following two properties hold:

- *Base Case.* $P(b)$ is a true statement for some specific b.
- *Induction Step.* If the statements $P(b), P(b+1), \ldots, P(n)$ are true then the statement $P(n+1)$ is also true.

Then we conclude that $P(n)$ is true **for all** $n \geqslant b$.

Sadly, this form of induction is still not strong enough to solve the exercise. Since the Fibonacci numbers are defined by the "second-order recurrence" $F_n = F_{n-1} + F_{n-2}$ we will need to check two base cases to get the induction started. Consider the statement

$$P(n) = \text{"}\varphi^{n-2} < F_n < \varphi^{n-1}\text{"}.$$

- *Base Cases.* Since $\varphi = (1+\sqrt{5})/2 \approx 1.61$ we observe that $P(3)$ and $P(4)$ are both true:

$$P(3) = \text{“}\varphi^1 < F_3 < \varphi^2\text{”} = \text{“}1.61 < 2 < 2.61\text{”}, \qquad \checkmark$$
$$P(4) = \text{“}\varphi^2 < F_4 < \varphi^3\text{”} = \text{“}2.61 < 3 < 4.24\text{”}. \qquad \checkmark$$

- *Induction Step.* Now fix some $n \geqslant 4$ and assume for induction that the statements $P(3), P(4), \ldots, P(n)$ are all true. In particular, we assume that the statements $P(n-1)$ and $P(n)$ are both true:

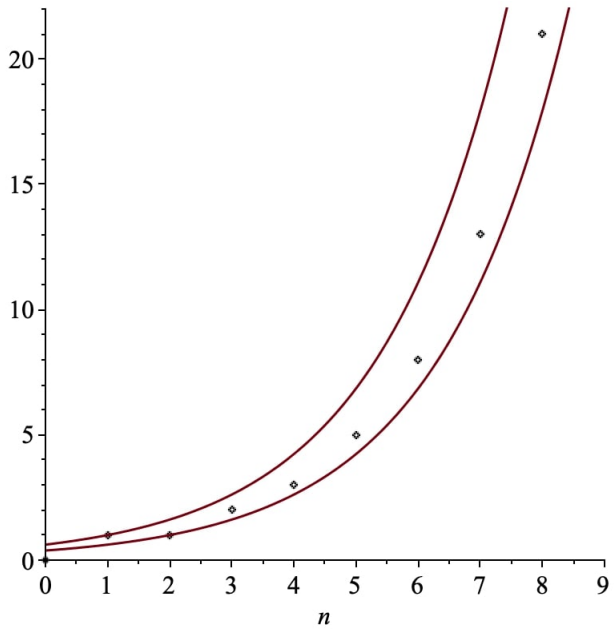
$$P(n-1) = \text{“}\varphi^{n-3} < F_{n-1} < \varphi^{n-2}\text{”}, \qquad \checkmark$$
$$P(n) = \text{“}\varphi^{n-2} < F_n < \varphi^{n-1}\text{”}. \qquad \checkmark$$

  By adding these two inequalities and using the identity $\varphi + 1 = \varphi^2$ we obtain

$$\begin{array}{ccccc} \varphi^{n-2} + \varphi^{n-3} & < & F_n + F_{n-1} & < & \varphi^{n-1} + \varphi^{n-2}, \\ \varphi^{n-3}(\varphi+1) & < & F_{n+1} & < & \varphi^{n-2}(\varphi+1), \\ \varphi^{n-3} \cdot \varphi^2 & < & F_{n+1} & < & \varphi^{n-2} \cdot \varphi^2, \\ \varphi^{n-1} & < & F_{n+1} & < & \varphi^n, \end{array}$$

  and hence $P(n+1)$ is also true.

Here is a picture of the first few Fibonacci numbers stuck between the curves φ^{n-1} and φ^{n-2}:



1.6. Use induction to verify the following formula:

$$1^3 + 2^3 + \cdots + n^3 = \frac{n^2(n+1)^2}{4} \qquad \text{for all } n \geqslant 1.$$

Solution. This one is straightforward algebra.

- *Base Case.* When $n = 1$ we have $1^3 = \frac{1^2 \cdot 2^2}{4}$.
- *Induction Step.* Fix some $n \geqslant 1$ and assume for induction that

$$1^3 + 2^3 + \cdots + n^3 \overset{\checkmark}{=} \frac{n^2(n+1)^2}{4}.$$

In this case we want to prove that

$$1^3 + 2^3 + \cdots + (n+1)^3 \overset{?}{=} \frac{(n+1)^2(n+2)^2}{4}.$$

To see this, we observe that

$$\begin{aligned} 1^3 + 2^3 + \cdots + (n+1)^3 &= 1^3 + 2^3 + \cdots + n^3 + (n+1)^3 \\ &= \frac{n^2(n+1)^2}{4} + (n+1)^3 \\ &= \frac{(n+1)^2}{4}\left[n^2 + 4(n+1)\right] \\ &= \frac{(n+1)^2}{4}\left[n^2 + 4n + 4\right] \\ &= \frac{(n+1)^2(n+2)^2}{4}. \end{aligned}$$

✓

Remark: It is a bit strange that

$$1^3 + 2^3 + \cdots + n^3 = (1 + 2 + \cdots + n)^2.$$

I don't know any good reason for this identity.

1.7. Let $\binom{n}{k}$ be the entry in the n-th row and k-th diagonal of Pascal's triangle. The *binomial theorem* tells us that for any number x and for any whole number $n \geqslant 0$ we have

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

Use this fact to simplify the following sums as much as possible:

$$\sum_{k=0}^{n} \binom{n}{k} = ? \qquad \text{and} \qquad \sum_{k=0}^{n} (-1)^k \binom{n}{k} = ?$$

Solution. Substituting $x = 1$ into the binomial theorem gives

$$(1+x)^n = \binom{n}{0} + \binom{n}{1} x + \binom{n}{2} x^2 + \cdots + \binom{n}{n} x^n,$$

$$(1+1)^n = \binom{n}{0} + \binom{n}{1}1 + \binom{n}{2}1^2 + \cdots + \binom{n}{n}1^n,$$
$$2^n = \binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n}.$$

Substituting $x = -1$ into the binomial theorem gives

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

We can use these formulas to investigate the Steiner-Schläfli Theorem when the dimension of the space is greater than or equal to the number of cuts. Suppose that we make n cuts of a d-dimensional space with $d \geqslant n$, and recall that we define $\binom{n}{k} = 0$ for all $k > n$. Then according to the theorem the maximum number of regions is

$$\sum_{k=0}^{d}\binom{n}{k} = \sum_{k=0}^{n}\binom{n}{k} = 2^n.$$

If $d \geqslant n$ then the maximum number of bounded regions is

$$\pm\sum_{k=0}^{d}(-1)^k\binom{n}{k} = \pm\sum_{k=0}^{n}(-1)^k\binom{n}{k} = 0.$$

In other words: If $n \leqslant d$ then it is possible to obtain 2^n regions from n cuts of d-dimensional space. If $n > d$ then the number of regions is less than 2^n. If $n \leqslant d$ then it is impossible to obtain a bounded region of d-dimensional space from n cuts.

Here are two extra challenge problems.

1.8. In Schläfli's book he actually states that the number of bounded regions is $\binom{n-1}{d}$. Prove that this agrees with the formula I stated above:

$$\binom{n-1}{d} = \binom{n}{d} - \binom{n}{d-1} + \binom{n}{d-2} - \cdots + (-1)^{d-1}\binom{n}{1} + (-1)^d\binom{n}{0}.$$

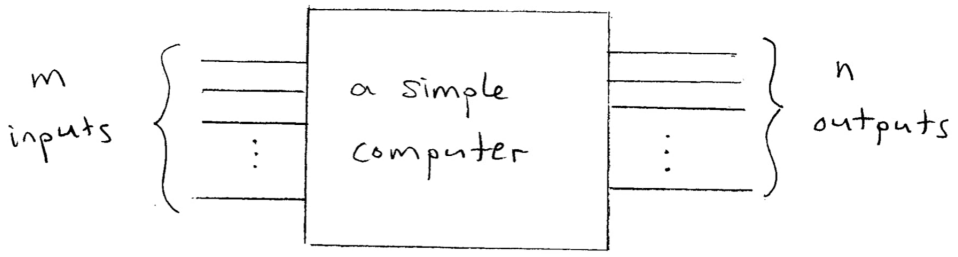
1.9. In Problem 1.1 we disovered the identity $\sum_{k=0}^{n}\binom{k+2}{2} = \binom{n+3}{3}$. Prove in general that

$$\sum_{k=0}^{n}\binom{k+d}{d} = \binom{n+d+1}{d+1} \qquad \text{for any } d \geqslant 0.$$

Wikipedia calls this the "hockey stick identity". The cases $d = 0, 1, 2$ are already known to us.

2 Boolean Algebra

To a first degree of approximation, an electronic computer is just a box with m wires going in and n wires coming out:



Each wire can only carry two signals: 0 (represented by low voltage) or 1 (represented by high voltage). Suppose that our computer is designed to add two numbers from the range 0 to 7. Using binary arithmetic, we could encode the input with 6 wires and the output with 4 wires. To compute "6 + 5" we would input 110101. Then the correct output would be 1011, which corresponds to "11". We will discuss the details in Section 2.4.

The attempt to build a mechanical (or electronic) computer was always tied to the philosophy of language. The mathematician and philosopher Wilhelm Gottfried Leibniz[9] had a lifelong dream to develop a "calculus of logic", to convert human thought into mechanical calculation. To this end he designed a mechanical calculator in 1672. He later also advocated the use of binary arithmetic based on the hexagrams of the ancient Chinese *I Ching*. Our modern notation for the calculus of logic is based on the works of George Boole from the 1850s.

In this chapter I will introduce the language of *Boolean algebra* and explain how it is applied to the design of simple computers. Boolean algebra comes in three basic flavors:

- set theory,
- logic,
- binary arithmetic.

We will treat these in order.

2.1 Set Theory

The language of set theory is the basic foundation for both logic and mathematics. Because the concept of a set is so fundamental we don't want to get too specific about it. (Mention Cantor.)

[9] Leibniz was a co-inventor of the Calculus along with Isaac Newton.

Definition of Sets

A *set* is a "collection of things". It has just one attribute, called *membership*. If S is a set we will use the notation

$$x \in S$$

to denote that "thing x is a member (or element) of the set S".[10]

A set with finitely many elements can be described using curly braces:

$$S = \{1, 2, 4, \text{apple}\}.$$

Observe that $1 \in S$ and $2 \in S$, but $3 \notin S$ and orange $\notin S$. The members of a set are not ordered:

$$\{1, 3, 2\} = \{1, 2, 3\}.$$

And sets do not care about repetition:

$$\{1, 3, 2, 3\} = \{1, 2, 3\}.$$

Indeed, we are only allowed to ask of a set whether the statement "$x \in S$" is true or false. There is no way to ask if an element is repeated, or if one element comes before another.

A set can have other sets as members, for example:

$$S = \{1, \{2\}, \{2, \{3, 4\}\}\}.$$

In this case we have $1 \in S$ but $2 \notin S$. However, we do have $\{2\} \in S$. Think of this as bags within bags. There is also a set corresponding to an empty bag. We call it the *empty set* and we write it like this:

$$\varnothing := \{\}.$$

Often we consider sets of numbers. Here are some of our favorites:[11]

- The set of *natural numbers*
$$\mathbb{N} = \{0, 1, 2, 3, 4, \ldots\}.$$
- The set of *integers*
$$\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}.$$
- The set of *rational numbers*
$$\mathbb{Q} = \left\{0, 1, -1, \frac{1}{2}, \frac{-1}{2}, 2, -2, 3, -3, \frac{1}{3}, \frac{-1}{3}, \frac{1}{4}, \frac{-1}{4}, \frac{2}{3}, \frac{-2}{3}, \ldots\right\}$$
It's okay if you don't see the pattern here. I'm sure you know what I'm talking about.

[10] The membership symbol "$\in$" is based on the Greek letter "ϵ", which stands for "element".

[11] $\mathbb{N}$ is for "natural", $\mathbb{Z}$ is for "Zahlen" (German for numbers), $\mathbb{Q}$ is for "quotients" and $\mathbb{R}$ is for "real".

- The set of *real numbers*
$$\mathbb{R} = \{0, 1, \sqrt{2}, e, \pi, \ldots\}.$$
Hmm. The set of real numbers is actually pretty hard to describe. I'm not going to define it in this class.

It is impossible to discuss sets for long without mentioning logic. The concept of "subset" is based on the words "if ... then".

Definition of Subsets

Given two sets A, B we say that A is a *subset* of B if for any thing x the following holds:

$$\text{if } x \in A \text{ then } x \in B.$$

In this case we will write $A \subseteq B$. I claim that $\varnothing \subseteq A$ is true for any set A. Does that make sense to you? It depends a bit on how you interpret the words "if ... then".

Example: Find all of the subsets of $\{1, 2, 3\}$.

Answer: There are $8 = 2^3$ subsets. In fact, we can interpret each subset as a "binary string of length 3". The symbol 1 in the i-th position means that we include i in the subset and the symbol 0 means that we do not include i. The number of binary strings is 2^3 because there are 2 choices for each symbol:

subset	binary string
$\{1, 2, 3\}$	111
$\{1, 2\}$	110
$\{1, 3\}$	101
$\{2, 3\}$	011
$\{1\}$	100
$\{2\}$	010
$\{3\}$	001
$\varnothing$	000

We can also use logical statements to implicitly define the elements of a subset.

Set Builder Notation

Let S be a set and for each element $x \in S$ let $P(x)$ be a logical statement. Then we

define the following subset of S:

$$\{x \in S : P(x)\} = \text{the set of } x \in S \text{ such that } P(x) \text{ is true.}$$

For example, here is how we define the set of "even numbers":

$$\{n \in \mathbb{Z} : n \text{ is a multiple of } 2\}.$$

To be more formal we can use the logical symbol $\exists$, which means "there exists":

$$\{n \in \mathbb{Z} : \exists k \in \mathbb{Z}, n = 2k\} = \{n \in \mathbb{Z} : \text{there exists some } k \in \mathbb{Z} \text{ such that } n = 2k\}$$

But now we have to be careful. With the definitions I just gave it is easy to write down nonsensical statements (also called paradoxes). The first such paradox was discovered by Bertrand Russell in 1901.

Russell's Paradox (1901)

Let S be the set of all sets and consider the following set:

$$R := \{A \in S : A \notin A\} = \text{the set of all } A \text{ such that } A \text{ is not a member of itself.}$$

Take a moment to convince yourself that the statement "$R \in R$" is complete nonsense.

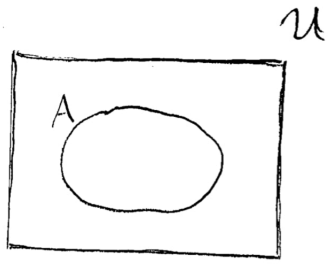
The lessons that we take from Russell's Paradox are the following:

- There is no set of all sets.
- A set cannot be an element of itself.

Unfortunately this makes the subject too technical for us. In order to avoid any complication we will adopt the following very strong assumption.

The Hypothesis of a Finite Universe

Let U be a fixed set with finitely many elements. From now on we will assume that every set A under discussion is a subset of U:



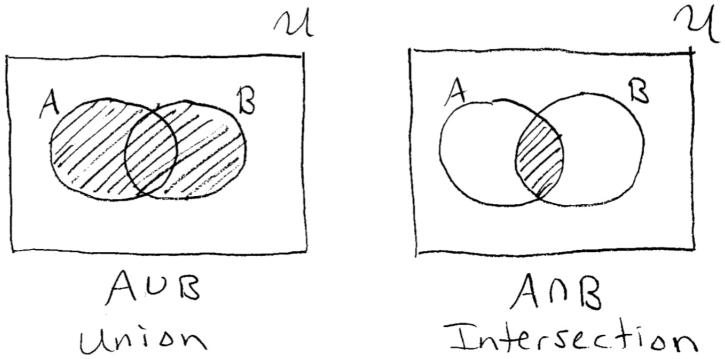
Now we can comfortably discuss the "Boolean algebra" of sets. For any two sets $A, B \subseteq U$ we define the *union*

$$A \cup B := \{x \in U : x \in A \text{ or } x \in B\} \subseteq U$$

and the *intersection*

$$A \cap B := \{x \in U : x \in A \text{ and } x \in B\} \subseteq U.$$

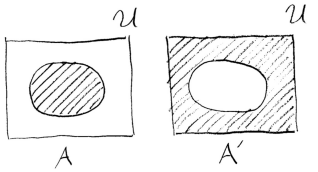
Here are some helpful pictures, called *Venn diagrams*:



For any set $A \subseteq U$ we also define the *complement*:

$$A' := \{x \in U : x \notin A\} \subseteq U.$$

The notation A' makes no sense unless we have a specific universal set in mind. Picture:



But do these pictures and formulas really define anything? Certainly a computer would not understand either language. So how would we explain the algebra of sets to a computer? We have a couple of options.

The first is to represent each subset as a binary string. Then the three operations $\cup, \cap, '$ are easy to formalize. For example, let $U = \{1, 2, 3, 4, 5, 6, 7\}$ with $A = \{1, 3, 5, 7\}$ and $B = \{4, 5, 6, 7\}$. Then the intersection is the "componentwise multiplication" of binary strings:

$$\begin{array}{ccccc} \{1,3,5,7\} & \cap & \{4,5,6,7\} & = & \{5,7\}, \\ 1010101 & \cap & 0001111 & = & 0000101. \end{array}$$

On the other hand, we could use a computer to encode the algebraic rules that the operations $\cup, \cap, '$ are supposed to satisfy.

Algebraic Properties of Sets (Boolean Algebra)

Let U be a finite universal set. Then for all subsets $A, B, C \subseteq U$ we have

(1) *Associative Laws*

$A \cup (B \cup C) = (A \cup B) \cup C$

$A \cap (B \cap C) = (A \cap B) \cap C$

(2) *Commutative Laws*

$A \cup B = B \cup A$

$A \cap B = B \cap A$

(3) *Algebraic Properties of $\varnothing$ and U*

$A \cup \varnothing = A$

$A \cap U = A$

(4) *Algebraic Properties of Complement*

$A \cup A' = U$

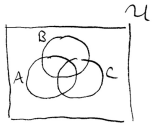
$A \cap A' = \varnothing$

(5) *Distributive Laws*

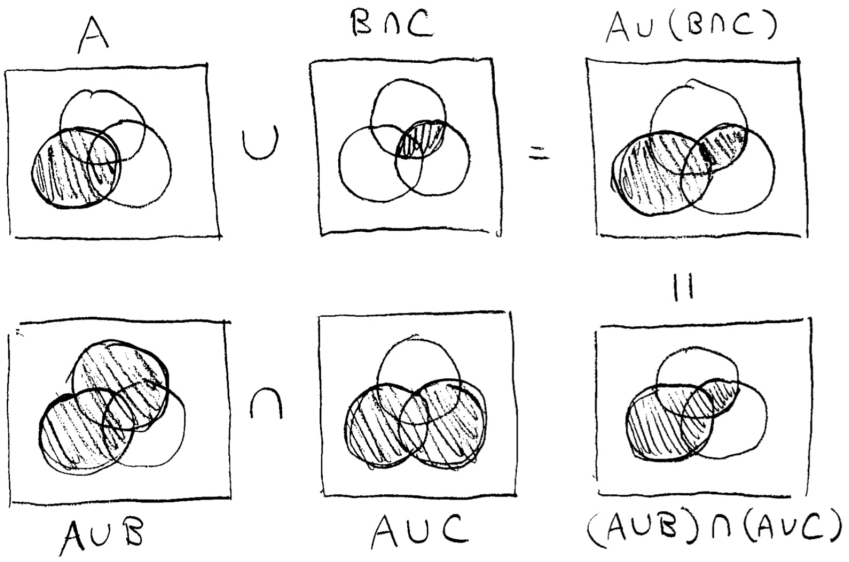
$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

The first four rules are mostly obvious, but we should use Venn diagrams to convince ourselves that the distributive laws are true. We can draw the sets $A, B, C \subseteq U$ as follows:



Then the first distributive law is proved by the following diagrams:



You should verify the second distributive law for yourself. I remember the distributive laws by thinking about addition and multiplication of numbers:

$$a \times (b + c) = (a \times b) + (a \times c).$$

However, this analogy is limited because there is a symmetry between $\cup$ and $\cap$ that does not exist between $+$ and $\times$. Indeed, the other distributive law for numbers is false:

$$a + (b \times c) \neq (a + b) \times (a + c).$$

There are infinitely many more algebraic properties of sets, but it turns out that any true formula involving the operations $\cup, \cap, '$ can be derived from the 5 basic rules above. In the next section we will discuss the important *de Morgan's Law*:

$$(A \cup B)' = A' \cap B' \qquad \text{and} \qquad (A \cap B)' = A' \cup B'.$$

2.2 Logic

It is impossible to discuss set theory without also discussing logic. For example, we already used the logical operations "FOR ALL", "IF... THEN", "OR", "AND" and "NOT" when we defined the basic operations on sets:

"$A \subseteq B$" = "FOR ALL $x \in U$, IF $x \in A$ THEN $x \in B$,"

$$A \cup B = \{x \in U : x \in A \text{ OR } x \in B\},$$
$$A \cap B = \{x \in U : x \in A \text{ AND } x \in B\},$$
$$A' = \{x \in U : \text{ NOT } x \in A\}.$$

You didn't get confused at the time because you thought I was speaking English. But these words in capital letters are not English at all. They are formal mathematical concepts that need to be defined precisely. We will do that in this section.

Just as *sets* are the basic objects of set theory, *statements* are the main objects of logic. Since these two concepts are so basic it is hard to define them precisely. Recall that I defined a set as a "collection of things". The definition of statements is similarly vague.

Definition of Statements

A *statement*[12] is a "sentence that has a definite truth value". To be more precise, we say that P is a statement if

$$P = T \text{ or } P = F, \text{ but not both.}$$

The symbols T and F are supposed to represent "true" and "false".

This definition necessarily restricts the domain of logic because most (all?) English sentences are not statements. Here are some examples:

- Let $n \in \mathbb{Z}$ be an integer. The sentence

  $$P = \text{"}n \text{ is even"}$$

  is a statement. I don't know if "n is even"$= T$ or "n is even"$= F$, but it is definitely one of them, and not both. You can think of P as a *Boolean variable*: $P \in \{T, F\}$.

- The following sentence is not a statement:

  "today is a nice day"

  You and I might agree that "today is a nice day"$= T$, but someone else might disagree with us and we have no basis on which to prove them wrong. Statements can only be discussed in a context where there are clear rules for deciding on the correct answer.

- The sentences "$1 + 2 = 3$" and "$1 + 2 = 4$" are both statements because

  $$\text{"}1 + 2 = 3\text{"} = T \qquad \text{and} \qquad \text{"}1 + 2 = 4\text{"} = F.$$

- What about this one?

[12] When we are being careful we will call this a *mathematical statement*, or a *logical statement*.

"this sentence is not a statement"

We will try to avoid sentences like this.

Now we are ready to define the basic operations of logic. If P and Q are statements, then the following expressions are also statements:

$$P \text{ OR } Q, \qquad P \text{ AND } Q, \qquad \text{NOT } P.$$

We just have to decide what their truth values should be for the different values of P and Q. Since $P \in \{T, F\}$ there are only $4 = 2^2$ possible ways to define NOT P:

P	(1)	(2)	(3)	(4)
T	T	T	F	F
F	T	F	T	F

Which of these most closely agrees with your understanding of the word "NOT"? I agree that the correct answer is (3). Furthermore, since $P, Q \in \{T, F\}$ there are exactly $16 = 2^4$ different ways to define the statement P OR Q:

P	Q	(1)	(2)	(3)	(4)	(5)	$\cdots$	(16)
T	T	T	T	T	T	T	$\cdots$	F
T	F	T	T	T	T	F	$\cdots$	F
F	T	T	T	F	F	T	$\cdots$	F
F	F	T	F	T	F	T	$\cdots$	F

Which of these most closely agrees with your understanding of the word "OR"? This time it will take you longer to decide, but eventually you will agree with me that the correct answer is (2). The expression P AND Q must also be defined by one of these 16 columns. After thinking for a bit we arrive at the following definition.

Definition of OR, AND, NOT **Using Truth Tables**

Let $P, Q \in \{T, F\}$ be statements. Then the statement NOT P is defined as follows:

P	NOT P
T	F
F	T

Furthermore, the statements P OR Q and P AND Q are defined as follows:

P	Q	P OR Q	P AND Q
T	T	T	T
T	F	T	F
F	T	T	F
F	F	F	F

These kinds of arrays are called *truth tables*. You can think of NOT as a "flipper", OR as a "T-detector", and AND as an "F-detector".

If we identify the logical operations OR, AND, NOT with the set operations $\cup, \cap, '$ then I claim that the algebra of logic satisfies exactly the same rules as the algebra of sets. As an exercise in these definitions, let us verify the second distributive law:

$$P \text{ AND } (Q \text{ OR } R) = (P \text{ AND } Q) \text{ OR } (P \text{ AND } R)$$

Since there are exactly $8 = 2^3$ ways to choose the values of $P, Q, R \in \{T, F\}$ our truth table must have 8 rows:[13]

P	Q	R	Q OR R	P AND $(Q$ OR $R)$	P AND Q	P AND R	$(P$ AND $Q)$ OR $(P$ AND $R)$
T	T	T	T	T	T	T	T
T	T	F	T	T	T	F	T
T	F	T	T	T	F	T	T
T	F	F	F	F	F	F	F
F	T	T	T	F	F	F	F
F	T	F	T	F	F	F	F
F	F	T	T	F	F	F	F
F	F	F	F	F	F	F	F

We observe that the fifth and eighth columns are equal.

As you can see, the word notations OR, AND, NOT get to be quite cumbersome. Therefore we will adopt the following symbolic notations from now on:

$$\text{OR} = \vee, \qquad \text{AND} = \wedge \qquad \text{and} \qquad \text{NOT} = \neg.$$

Let us return to de Morgan's Law using this new language.

De Morgan's Law

[13] The ordering of the rows doesn't matter but you should try to be consistent.

For all statements P and Q I claim that

$$\neg(P \vee Q) = \neg P \wedge \neg Q \qquad \text{and} \qquad \neg(P \wedge Q) = \neg P \vee \neg Q.$$

You will prove these identities on the homework using a truth table. But I claim that the identities are obvious if we look at them from a certain point of view. The key is to first generalize the statement by induction. For any sequence of statements $P_1, P_2, \ldots, P_n$ I claim that the following properties hold:

$$\neg(P_1 \vee P_2 \vee \cdots \vee P_n) = \neg P_1 \wedge \neg P_2 \wedge \cdots \wedge \neg P_n,$$
$$\neg(P_1 \wedge P_2 \wedge \cdots \wedge P_n) = \neg P_1 \vee \neg P_2 \vee \cdots \vee \neg P_n.$$

Proof by Induction. The proof is the same for both statements, so we will only prove the first. Let's say that the base case is $n = 2$, which you have already proved. Now fix some $n \geqslant 2$ and assume for induction that

$$\neg(P_1 \vee P_2 \vee \cdots \vee P_n) = \neg P_1 \wedge \neg P_2 \wedge \cdots \wedge \neg P_n.$$

For convenience let us define $Q := P_1 \vee P_2 \vee \cdots \vee P_n$. Then we also have

$$\begin{aligned}
\neg(P_1 \vee P_2 \vee \cdots \vee P_{n+1}) &= \neg(Q \vee P_{n+1}) \\
&= \neg Q \wedge \neg P_{n+1} && \text{case } n = 2 \\
&= \neg(P_1 \vee P_2 \vee \cdots \vee P_n) \wedge \neg P_{n+1} \\
&= \neg P_1 \wedge \neg P_2 \wedge \cdots \wedge \neg P_n \wedge \neg P_{n+1}.
\end{aligned}$$

□

Let's think a bit about the compound statements:

$$\bigvee_{i=1}^{n} P_i := P_1 \vee P_2 \vee \cdots \vee P_n,$$
$$\bigwedge_{i=1}^{n} P_i := P_1 \wedge P_2 \wedge \cdots \wedge P_n.$$

I claim that these are easier than they look. Indeed, since OR $= \vee$ is a "T-detector", the compound statement

$$P_1 \text{ OR } P_2 \text{ OR } \cdots \text{ OR } P_n = P_1 \vee P_2 \vee \cdots \vee P_n$$

has the value T when **at least one of the statements** P_i is true. We have a special notation for this, called the *existential quantifier* $\exists$:

$$\bigvee_{i=1}^{n} P_i = (\exists i \in \{1, 2, \ldots, n\}, P_i)$$

$$= \text{“there exists some } i \in \{1, 2, \ldots, n\} \text{ such that } P_i \text{ is true”}.$$

Similarly, since AND $= \wedge$ is an “F-detector”, the compound statement

$$P_1 \,\text{AND}\, P_2 \,\text{AND}\, \cdots \,\text{AND}\, P_n = P_1 \wedge P_2 \wedge \cdots \wedge P_n$$

has the value T only if **all of the statements** P_i are true. To express this we use the *universal quantifier* $\forall$:

$$\begin{aligned}\bigwedge_{i=1}^{n} P_i &= (\forall i \in \{1, 2, \ldots, n\}, P_i) \\ &= \text{“for all } i \in \{1, 2, \ldots, n\} \text{ the statement } P_i \text{ is true”}.\end{aligned}$$

And what does this have to with de Morgan’s Law? By rewriting the extended de Morgan’s Law in terms of the quantifiers $\exists$ and $\forall$ we obtain the following:

$$\begin{aligned}\neg\left(\bigvee_{i=1}^{n} P_i\right) &= \bigwedge_{i=1}^{n} \neg P_i, \\ \neg\left(\exists i \in \{1, \ldots, n\}, P_i\right) &= (\forall i \in \{1, \ldots, n\}, \neg P_i), \\ \text{“there does not exist any } i \text{ such that } P_i \text{ is true”} &= \text{“}P_i \text{ is false for all } i\text{”}.\end{aligned}$$

The other version says:

$$\begin{aligned}\neg\left(\bigwedge_{i=1}^{n} P_i\right) &= \bigvee_{i=1}^{n} \neg P_i, \\ \neg\left(\forall i \in \{1, \ldots, n\}, P_i\right) &= (\exists i \in \{1, \ldots, n\}, \neg P_i), \\ \text{“it is not the case that } P_i \text{ is true for all } i\text{”} &= \text{“there exists some } i \text{ such that } P_i \text{ is false”}.\end{aligned}$$

When you put it like that, de Morgan’s Law seems pretty obvious. The following definition formalizes this intuition.

Quantifiers and de Morgan’s Law

Let S be a set and for each element $x \in S$ let $P(x)$ be a statement. Then we define the *existential* ($\exists$) and *universal* ($\forall$) *quantifiers* as follows:

$$\begin{aligned}[\exists x \in S, P(x)] &= \text{“there exists some } x \in S \text{ such that } P(x) \text{ is true”}, \\ [\forall x \in S, P(x)] &= \text{“}P(x) \text{ is true for all } x \in S\text{”},\end{aligned}$$

Then we have the following abstract version of *de Morgan’s Law*:

$$\neg\left[\exists x \in S, P(x)\right] = \left[\forall x \in S, \neg P(x)\right],$$

$$\neg\left[\forall x \in S, P(x)\right] = \left[\exists x \in S, \neg P(x)\right].$$

We have now defined five logical operators:

$$\text{OR} = \vee, \quad \text{AND} = \wedge, \quad \text{NOT} = \neg, \quad \text{EXISTS} = \exists, \quad \text{FORALL} = \forall.$$

And we have discussed the "algebraic relations" among them; namely, the five basic rules of Boolean algebra together with de Morgan's Law ($\neg\exists = \forall\neg$). But we still have not formalized the words "IF ... THEN". I will just tell you the definition and then try to explain it.

Definition of Material Implication

Let P and Q be statements. Then we define a new statement $P \Rightarrow Q$ with the following truth table:

P	Q	$P \Rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

In words, we will read this as

$$P \Rightarrow Q = \text{"IF } P \text{ THEN } Q\text{"} = \text{"}P \text{ IMPLIES } Q\text{"}.$$

Don't hurt your brain trying to make sense of this in terms of English. I think it only makes sense if we translate the definition into set theory. Let $A, B \subseteq U$ be subsets of the universal set. Recall that the notion of "subset" is defined in terms of the universal quantifier ($\forall$) and material implication ($\Rightarrow$):

$$\begin{aligned} \text{"}A \subseteq B\text{"} &= \text{"FORALL } x \in U, x \in A \text{ IMPLIES } x \in B\text{"}, \\ \text{"}A \subseteq B\text{"} &= \text{"}\forall x \in U, x \in A \Rightarrow x \in B\text{"}. \end{aligned}$$

Then applying de Morgan's Law gives

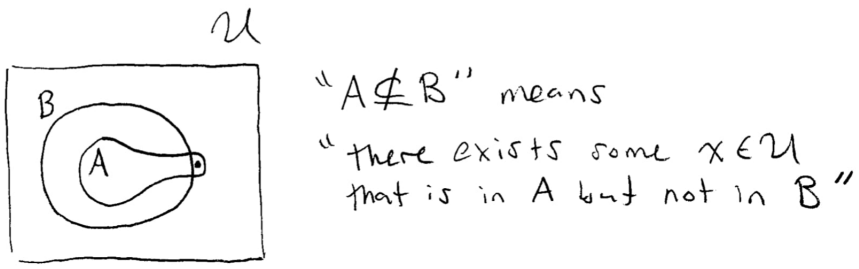
$$\begin{aligned} \neg\text{"}A \subseteq B\text{"} &= \text{"}\forall x \in U, x \in A \Rightarrow x \in B\text{"}, \\ \text{"}A \nsubseteq B\text{"} &= \text{"}\exists x \in U, \neg(x \in A \Rightarrow x \in B)\text{"}. \end{aligned}$$

On the other hand, it is pretty clear that we should have

$$\text{"}A \nsubseteq B\text{"} = \text{"EXISTS } x \in U, x \in A \text{ AND NOT } x \in B)\text{"},$$

$$\text{“}A \nsubseteq B\text{”} = \text{“}\exists x \in U, x \in A \wedge \neg\, x \in B\text{”}.$$

Indeed, consider the following diagram:



By comparing the two expressions for “$A \nsubseteq B$” we must have

$$\neg(x \in A \Rightarrow x \in B) = x \in A \wedge \neg\, x \in B,$$

and more generally for any two statements P, Q we should have

$$\begin{aligned} \neg(P \Rightarrow Q) &= P \wedge \neg Q, \\ P \Rightarrow Q &= \neg(P \wedge \neg Q) \\ &= \neg P \vee \neg(\neg Q) && \text{de Morgan's Law} \\ &= \neg P \vee Q. \end{aligned}$$

Does this agree with the definition of $P \Rightarrow Q$ that I gave above? Let's check the truth table:

P	Q	$\neg P$	$\neg P \vee Q$	$P \Rightarrow Q$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Yes it does. I don't ask you to feel that this is true.[14] I just ask you to memorize the Boolean definition of implication:

$$\boxed{P \Rightarrow Q := \neg P \vee Q.}$$

2.3 Functions

We have seen the definition of the “logical connectives” OR, AND, NOT in terms of truth tables, but in order to apply these to the design of computers we need to think of them as certain kinds of *functions*. In order to give the formal definition I must first define a new construction on sets.

[14] The definition of implication is the place where formal logic diverges most from natural languages such as English. The biggest controversy has to do with the implications $(F \Rightarrow T) = T$ and $(F \Rightarrow F) = T$. In practice these will never come up.

Definition of the Cartesian Product of Sets

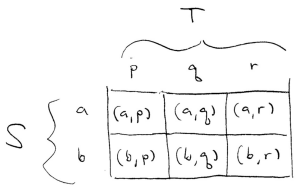
Let S and T be sets. For any two elements $s \in S$ and $t \in T$ we may consider the *ordered pair* (s, t). This is sort of like a set with two elements, but where order matters. Then we define the *Cartesian product* of S and T as the set of ordered pairs:

$$S \times T := \{(s, t) : s \in S \text{ and } t \in T\}.$$

For example, let $S = \{a, b\}$ and $T = \{p, q, r\}$. Then the Cartesian product is given by

$$S \times T = \{(a, p), (a, q), (a, r), (b, p), (b, q), (b, r)\}.$$

Sometimes it is more meaningful to think of this set as a "rectangular array":



If S and T are finite sets, then by counting the cells in the rectangle we obtain

$$\#(S \times T) = \#S \cdot \#T.$$

This explains why we call it the Cartesian **product**. Now I can give the formal definition of a function.

Definition of Functions

Let S and T be sets and let us think of each element $(s, t) \in S \times T$ of the Cartesian product as an "arrow" from s to t. Let $f \subseteq S \times T$ be a set of arrows and consider the following four properties:

(1) For each $s \in S$ there exists **at most one** $t \in T$ such that $(s, t) \in f$.

(2) For each $s \in S$ there exists **at least one** $t \in T$ such that $(s, t) \in f$.

(3) For each $t \in T$ there exists **at most one** $s \in S$ such that $(s, t) \in f$.

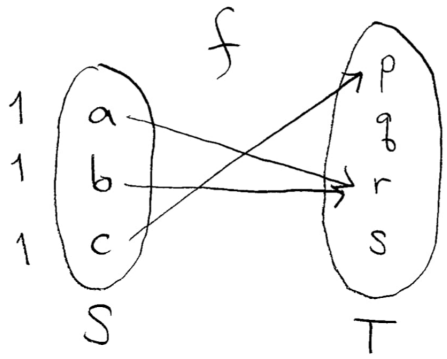
(4) For each $t \in T$ there exists **at least one** $s \in S$ such that $(s, t) \in f$.

We say that f is a *function* if it satisfies (1) and (2). In this case, for each $s \in S$ there exists **exactly one** $t \in T$ such that $(s, t) \in f$. Since this element t is unique we will give it a special name:

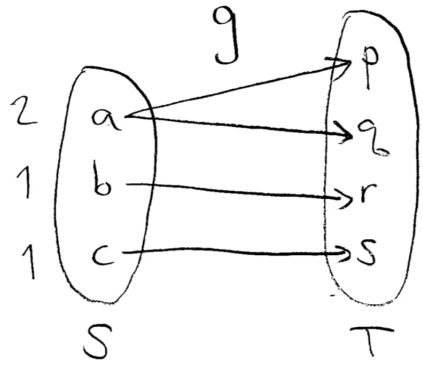
$$f(s) = t \qquad \Longleftrightarrow \qquad (s, t) \in f.$$

If $f \subseteq S \times T$ is a function then we will write $f : S \to T$. If f (1),(2),(3) we call it an *injective function* (or one-to-one). If f satisfies (1),(2),(4) we call it a *surjective function* (or onto). Finally, if f satisfies all four properties (1), (2), (3), (4) then we call it a *bijective function* (or a one-to-one correspondence).

That definition is a lot to unpack, so let us look at some examples. First consider the sets $S = \{a, b, c\}$ and $T = \{p, q, r, s\}$. Then the following set of arrows $f \subseteq S \times T$ is a function:



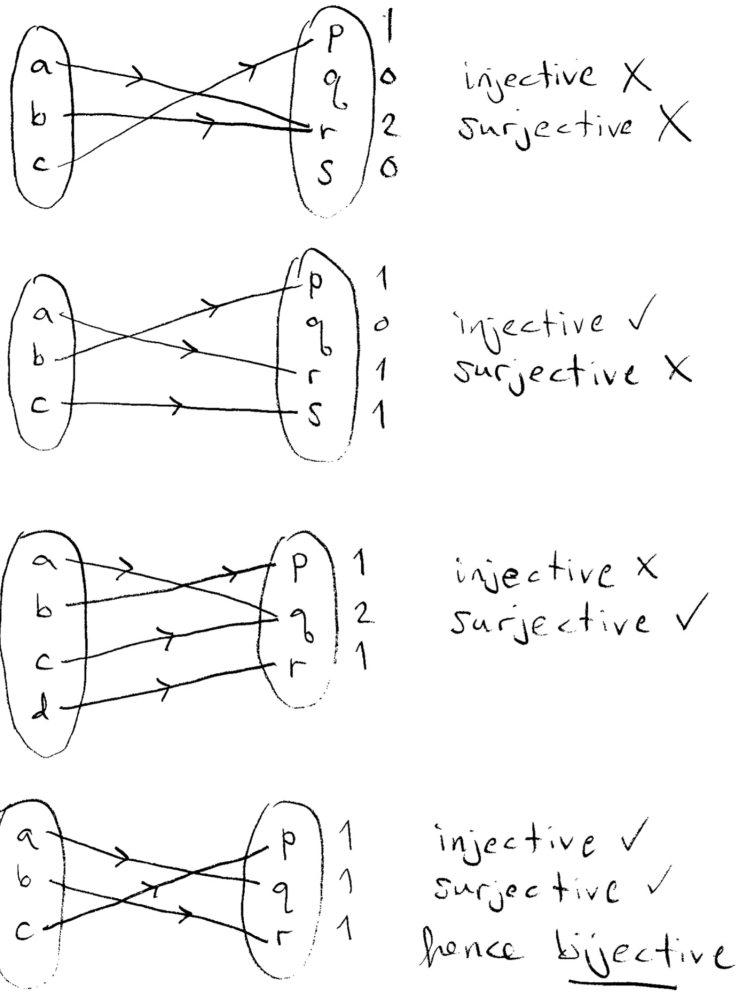
Indeed, the numbers on the left show that each element of S has exactly one arrow. This allows us to write $f(a) = r$, $f(b) = r$ and $f(c) = p$. However, the following set of arrows $g \subseteq S \times T$ is **not** a function:



In this case g satisfies (2) but it does not satisfy (1). Indeed, the 2 on the left indicates that there are two arrows coming out of a. This means that we cannot define $g(a)$ uniquely.

Now let's look at different kinds of functions. In order to determine if a function $f : S \to T$ is injective or surjective we count the arrows coming into each element of T. If all of the

numbers are $\leqslant 1$ then the function is injective. If all of the numbers are $\geqslant 1$ then the function is surjective. If all of the numbers are $= 1$ then the function is bijective:



You may have noticed that the size of the sets puts restrictions on the kinds of functions that can exist. You will prove the following properties on the homework:

- If there exists an injection $S \to T$ then $\#S \leqslant \#T$.
- If there exists a surjection $S \to T$ then $\#S \geqslant \#T$.
- If there exists a bijection $S \to T$ then $\#S = \#T$.

Sometimes this is a very convenient way to prove that two sets have the same size. For example, on the homework you will prove that for any set S there exists a bijection between the following two sets:

$$\{\text{subsets of } S\} \longleftrightarrow \{\text{functions } S \to \{T, F\}\}.$$

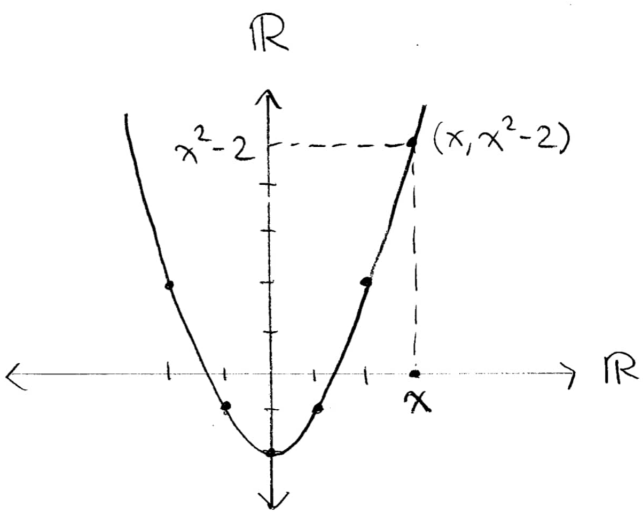
Then you will use this bijection to conclude that

$$\#\{\text{subsets of } S\} = \#\{\text{functions } S \to \{T, F\}\} = 2^{\#S}.$$

Sometimes the Cartesian product set $S \times T$ can be visualized. In this case we can visualize a function $f : S \to T$ as a subset of $S \times T$. For example, let $\mathbb{R}$ be the set of real numbers and let $f : \mathbb{R} \to \mathbb{R}$ be the function defined by the formula $f(x) := x^2 - 2$. Then each "arrow" $(x, x^2 - 2)$ of the function corresponds to a "point" in the Cartesian plane $\mathbb{R}^2 := \mathbb{R} \times \mathbb{R}$:

$$f = \{(x, x^2 - 2) : x \in \mathbb{R}\} \subseteq \mathbb{R} \times \mathbb{R}.$$

You are probably familiar with this set as the *graph* of the function. In this case the graph looks like a parabola:



If $f \subseteq \mathbb{R} \times \mathbb{R}$ is an arbitrary set of points in the Cartesian plane then we can rephrase the four properties as follows:

(1) Each vertical line intersects f in at most one point.

(2) Each vertical line intersects f in at least one point.

(3) Each horizontal line intersects f in at most one point.

(4) Each horizontal line intersects f in at least one point.

If (1) and (2) hold (called the "vertical line test") then f is the graph of a function. The "horizontal line test" tells us if this function is injective or surjective. In the case of $f = \{(x, x^2 - 2) : x \in \mathbb{R}\}$ we see that the function is **not injective** because many horizontal lines intersect the graph in two points. Furthermore, this function is **not surjective** because many

horizontal lines intersect the graph in zero points.[15] If follows that the function $f(x) = x^2 - 2$ is not invertible. We can generalize this situation as follows.

Existence of Inverse Functions

Let S and T be sets and consider a function with graph $f \subseteq S \times T$. We define the "inverse graph" by reversing all the arrows:

$$f^{-1} := \{(t, s) : (s, t) \in f\} \subseteq T \times S.$$

But the set f^{-1} is not always the graph of a function $T \to S$. I claim that

$$f^{-1} \text{ is the graph of a function} \iff f \text{ is the graph of a } \textbf{bijective} \text{ function.}$$

In other words, the function f is invertible if and only if it is bijective.

Proof. Since f is a function $S \to T$ we know that (1) and (2) hold. Then we have

$$\begin{aligned} f^{-1} \text{ is a function} &\iff (3) \text{ and } (4) \text{ hold} \\ &\iff f \text{ is bijective.} \end{aligned}$$

□

Finally, let us return to the "logical connectives" OR, AND, NOT. These are examples of functions defined between sets of the form

$$\begin{aligned} \{T, F\}^n &= \underbrace{\{T, F\} \times \{T, F\} \times \cdots \times \{T, F\}}_{n \text{ times}} \\ &= \{\text{ordered words of length } n \text{ from the alphabet } \{T, F\}\}. \end{aligned}$$

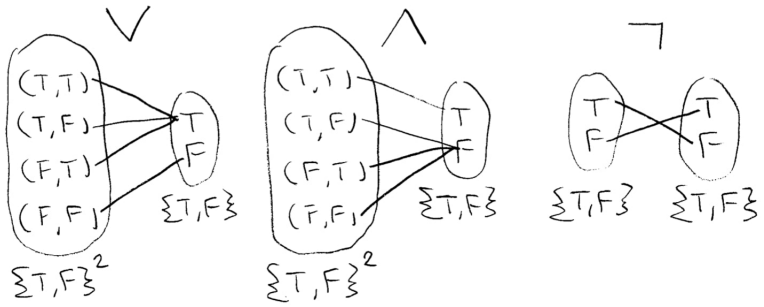
Definition of Boolean Functions

A *Boolean function* with m inputs and n outputs is defined as a function from the set $\{T, F\}^m$ to the set $\{T, F\}^n$:

$$f : \{T, F\}^m \to \{T, F\}^n.$$

[15] Sometimes we improve the properties of a graph by restricting the domain and codomain. For example, "tangent function" $\tan : \mathbb{R} \to \mathbb{R}$ is not actually a function because it is not defined when $x = \pi/2 + k\pi$. It becomes a function (in fact, a bijective function) when we restrict the domain to $\tan : (-\pi/2, \pi/2) \to \mathbb{R}$. The function $\sin : \mathbb{R} \to \mathbb{R}$ is not surjective or injective. We can make it both by restricting the domain and the codomain to $\sin : [-\pi/2, \pi/2] \to [-1, 1]$.

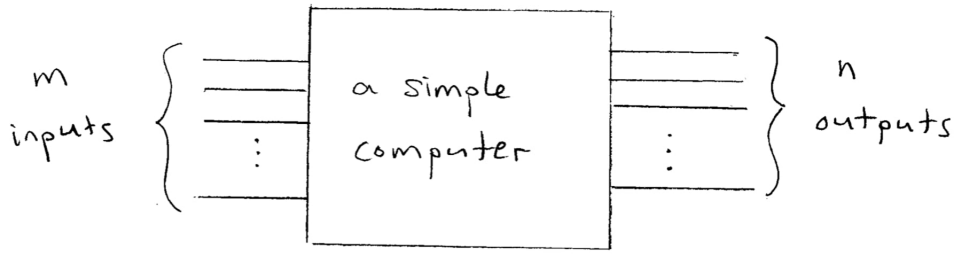
For example, here are our three favorite Boolean functions:



And what does the graph of a Boolean function look like? In fact, the graph of a Boolean function is just a truth table. Think about it.

2.4 Logic Circuits

At the beginning of this chapter I drew a picture of a simple computer:



Now I can tell you that a simple computer is the same thing as a *Boolean function* with m inputs and n outputs. However, instead of the set $\{T, F\}$ to represent true and false, we use the set $\{1, 0\}$ to represent high voltage and low voltage.

In this section I will show you how to build a useful computer. The simplest thing we might want to do is add two numbers. Let $x, y \in \{0, 1\}$ be any two "1-bit numbers". Then we have the following table:

x	y	$x + y$
1	1	?
1	0	1
0	1	1
0	0	0

Unfortunately, there is no amount of voltage that can represent the number 2. Instead we will use the following scheme, which was advocated by Leibniz in the early 1700s.

Binary Expansion

Let $n \geqslant 0$ be any non-negative integer. Then there exists a unique sequence of numbers $r_0, r_1, r_2, \ldots$ from the set $\{0, 1\}$ such that

$$n = \sum_{i=0}^{\infty} r_i \cdot 2^i = r_0 + 2r_1 + 4r_2 + 8r_3 + \cdots .$$

In this case we will write

$$n = (\cdots r_3 r_2 r_1 r_0)_2$$

and call this the *binary expansion of* n. (Compare this to the usual *decimal expansion*, which uses the base 10 instead of 2.) The number r_i is called the *i-th bit* of n (short for *binary digit*). As with the decimal notation, we can stop after the highest non-zero bit.

Proof. We will prove a more general result in the next chapter. ▫

For example, here are the binary expansions of the numbers 0 through 7. Maybe you recognize these expansions from our constant use of truth tables:

number	binary expansion
7	$(111)_2$
6	$(110)_2$
5	$(101)_2$
4	$(100)_2$
3	$(011)_2$
2	$(010)_2$
1	$(001)_2$
0	$(000)_2$

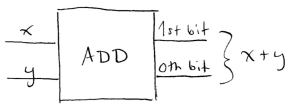
From now on we will omit the parentheses and the subscript 2 from the notation. Now we see that the sum of two "1-bit numbers" can be a "2-bit number":

$$1 + 1 = 10.$$

Here is the full table:

x	y	$x + y$
1	1	10
1	0	01
0	1	01
0	0	00

Technically speaking, this table is the graph of a binary function $+ : \{0,1\}^2 \to \{0,1\}^2$ with 2 inputs and 2 outputs. Here is a picture:



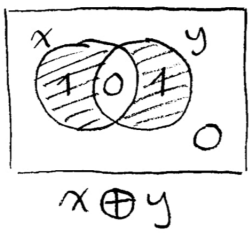
Now we want to build this computer. In other words, we want to find a Boolean expression for each bit of the output. The first bit is easy; it's just $x \wedge y$, which can also be thought of as *binary multiplication*:

x	y	$x \wedge y$	xy	1st bit of $x + y$
1	1	1	1	1
1	0	0	0	0
0	1	0	0	0
0	0	0	0	0

The second bit is harder. It can be thought of as *binary addition*, or *addition mod* 2. We will denote this with the special symbol $\oplus$:

x	y	$x \oplus y$	2nd bit of $x + y$
1	1	0	0
1	0	1	1
0	1	1	1
0	0	0	0

But we would prefer to express this in terms of the standard functions $\vee, \wedge, \neg$. We can do this by thinking of the truth table as a Venn diagram:



If we think of x and y as sets, then the two shaded regions are $x \cap y'$ and $x' \cap y$, hence the name of the shaded area is $(x \cap y') \cup (x' \cup y)$. Alternatively, we can express this region as $(x \cup y) \cap (x \cap y)'$. Thus we have the following two expressions:

$$\begin{aligned} x \oplus y &= (x \wedge \neg y) \vee (\neg x \wedge y) \\ &= (x \vee y) \wedge \neg(x \wedge y). \end{aligned}$$

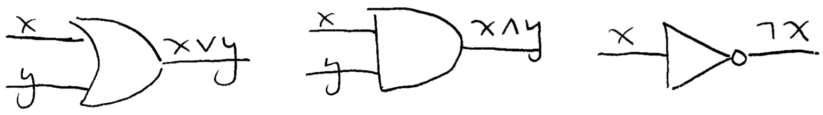
In terms of logic we call this the *exclusive or function*:

$$\begin{aligned} P \,\text{XOR}\, Q &:= (P \,\text{AND NOT}\, Q) \text{ OR } (Q \,\text{AND NOT}\, P) \\ &:= (P \text{ OR } Q) \,\text{AND NOT}\, (P \,\text{AND}\, Q) \\ &= \text{“}P \text{ or } Q \text{ but not both”}. \end{aligned}$$

Next let me show you how to draw pictures of these functions.

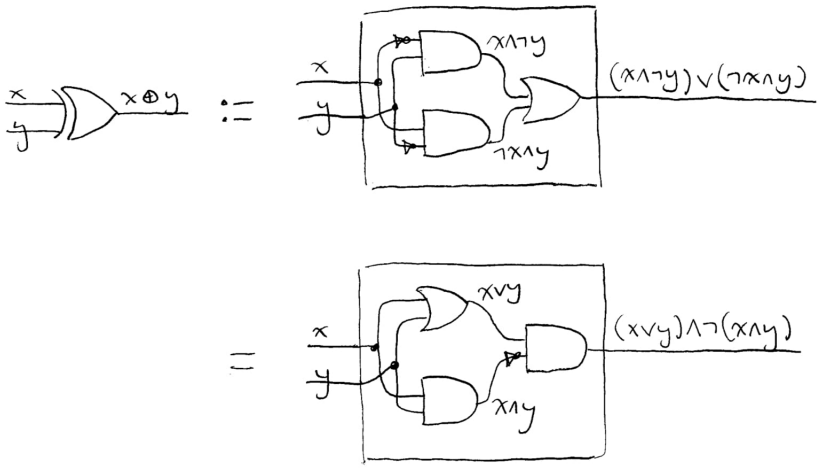
Pictures for Boolean Functions (Logic Gates)

We have the following pictures for the three basic Boolean functions:

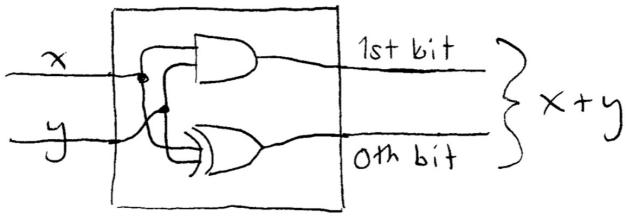


These pictures are called *logic gates*. Presumably and electrical engineer would know how to build physical versions of these.

We can wire together logic gates to create a picture of any Boolean function. For example, here are two different pictures of the XOR function:



Pictures of this kind are called *logic circuits*. Finally, we can draw a logic circuit for the *binary adder*:



We have designed our first computer, but it is pretty humble. Let me end this section by showing you how to add three bits instead of two! Since the largest sum of three bits is $1 + 1 + 1 = 11$, this will be a Boolean function with 3 inputs and 2 outputs:

x	y	z	$x + y + z$
1	1	1	11
1	1	0	10
1	0	1	10
1	0	0	01
0	1	1	10
0	1	0	01
0	0	1	01
0	0	0	00

Here are Venn diagrams for the 1st bit and the 0th bit of $x + y + z$:

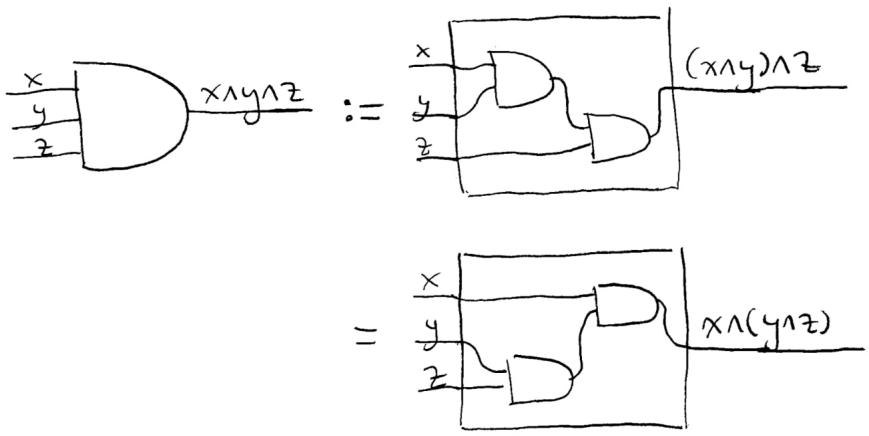
There are many ways to express these in terms of $\vee, \wedge, \neg$ but the easiest method is called the *disjunctive normal form*. Here we just name the shaded regions and then sum them up:[16]

$$\text{1st bit} = (x \wedge y \wedge z) \vee (x \wedge y \wedge \neg z) \vee (x \wedge \neg y \wedge z) \vee (\neg x \wedge y \wedge z),$$

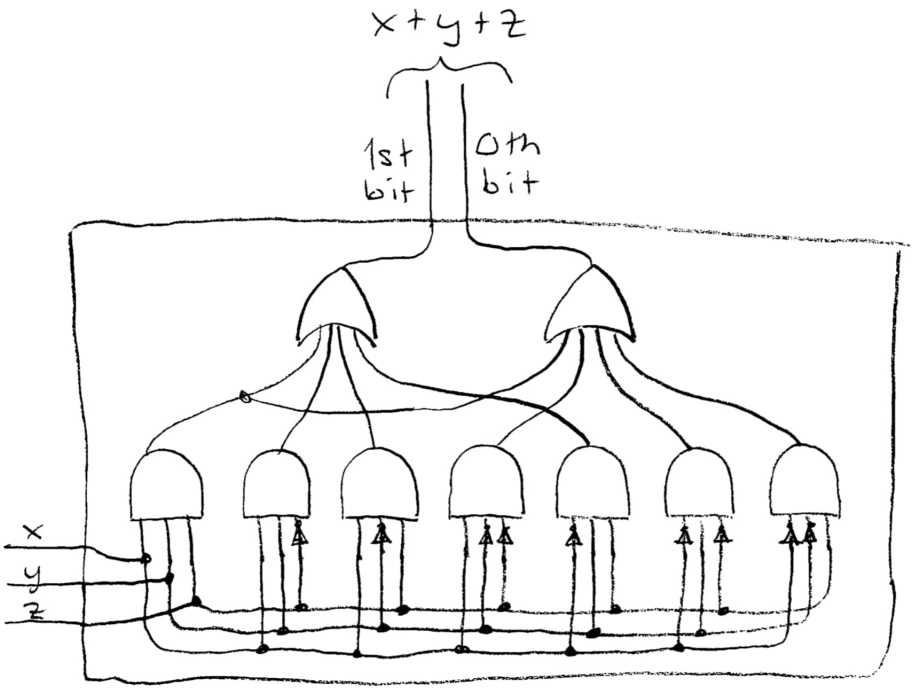
[16]The name "disjunctive normal form" comes from the fact that $\vee$ is sometimes called "disjunction". There is also a "conjunctive normal form" which expressed the unshaded regions of the Venn diagram.

$$\text{2nd bit} = (x \wedge y \wedge z) \vee (x \wedge \neg y \wedge \neg z) \vee (\neg x \wedge y \wedge \neg z) \vee (\neg x \wedge \neg y \wedge z).$$

In order to simplify the diagram we will use the fact that $\vee$ and $\wedge$ are associative operations to define extended AND and OR gates. For example, here is an AND gate with three inputs:



Finally, we can draw a logic circuit for the sum of three bits:



On the homework you will draw a more efficient version of this circuit.

2.5 Abstract Boolean Algebra

We have seen three languages that satisfy the same abstract properties:

- the algebra of sets,
- the algebra of logic,
- binary arithmetic.

The contribution of George Boole in the 1850s was to recognize the common features of these three languages. Within a few generations, other mathematicians such as Charles Saunders Peirce and Ernst Schröder formalized Boole's ideas to obtain the following definition.

Definition of Abstract Boolean Algebra

A *Boolean algebra* is a set B together with three functions

$\vee : B \times B \to B$ called *join*,

$\wedge : B \times B \to B$ called *meet*,

$\neg : B \to B$ called *complement*,

and two special elements $0 \neq 1 \in B$ called *zero* and *one*, which satisfy the following rules:

(1) *Associaive Laws.* For all $, a, b, c \in B$ we have

$$a \wedge (b \wedge c) = (a \wedge b) \wedge c$$

$$a \vee (b \vee c) = (a \vee b) \vee c$$

(2) *Commutative Laws.* For all $a, b \in B$ we have

$$a \vee b = b \vee a$$

$$a \wedge b = b \wedge a$$

(3) *Properties of* 0 *and* 1. For all $a \in B$ we have

$$a \vee 0 = a$$

$$a \wedge 1 = a$$

(4) *Properties of Complement.* For all $a \in B$ we have

$$a \vee \neg a = 1$$

$$a \wedge \neg a = 0$$

(5) *Distributive Properties.* For all $a, b, c \in B$ we have

$$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$$
$$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$$

What are the advantages of this very abstract definition? There are at least two. The first advantage is that computers don't understand human things like Venn diagrams and logical arguments. The language of Boolean algebra is purely formal and easy to teach to a computer. The second advantage is that it allows us humans to make fewer mistakes by converting the analysis of arguments into the mechanical manipulation of symbols.

Beginning with the five axioms, we can prove other properties, called theorems. A *theorem* is any true equation that can be obtained by successively applying the axioms. To end this chapter I will show you a formal proof of de Morgan's Laws. In order to shorten the process let me make first make the following observation.

The Duality Principle

We observe that the five rules of Boolean algebra remain the same after we switch

$$\vee \leftrightarrow \wedge \qquad \text{and} \qquad 0 \leftrightarrow 1.$$

Thus any theorem obtained from the axioms is still true after we make these switches. This gives us two theorems for the price of one proof.

(6) *Idempotence.* For all $a \in B$ we have

$$a \vee a = a$$
$$a \wedge a = a$$

Proof. In view of the Duality Principle, we need only prove the first statement. At each step I will quote the rule that I used:

$$\begin{aligned} a &= a \vee 0 && (3)\\ &= a \vee (a \wedge \neg a) && (4)\\ &= (a \vee a) \wedge (a \vee \neg a) && (5)\\ &= (a \vee a) \wedge 1 && (4)\\ &= a \vee a && (3) \end{aligned}$$

□

(7) *Complementarity of* 0 *and* 1. We have

$\neg 0 = 1$

$\neg 1 = 0$

Proof. Again, we only need to prove the first statement:

$$\begin{aligned} \neg 0 &= \neg 0 \vee 0 && (3)\\ &= 0 \vee \neg 0 && (2)\\ &= 1 && (4) \end{aligned}$$

□

(8) *I don't have a good name for this.* For all $a \in B$ we have

$a \vee 1 = 1$

$a \wedge 0 = 0$

Proof. We have

$$\begin{aligned} a \vee 1 &= a \vee (a \vee \neg a) && (4)\\ &= (a \vee a) \vee \neg a && (1)\\ &= a \vee \neg a && (6)\\ &= 1 && (4) \end{aligned}$$

□

Notice that we used (6) in the proof of (8). That's okay. Each theorem is considered as a new rule that we can use in future proofs.

(9) *Absorption Properties.* For all $a, b \in B$ we have

$a \vee (a \wedge b) = a$

$a \wedge (a \vee b) = a$

Proof. We have

$$\begin{aligned} a \vee (a \wedge b) &= (a \wedge 1) \vee (a \wedge b) && (3)\\ &= a \wedge (1 \vee b) && (5)\\ &= a \wedge 1 && (2), (8)\\ &= a && (3) \end{aligned}$$

□

(10) *Cancellation.* For all $a, b, c \in B$ we have

$$a \wedge c = b \wedge c \text{ and } a \vee c = b \vee c \text{ imply } a = b.$$

Proof. To begin, we assume that $a \wedge c = b \wedge c$ and $a \vee c = b \vee c$. It follows that

$$\begin{aligned}
a &= a \vee (a \wedge c) && (9)\\
&= a \vee (b \wedge c) && \text{by assumption}\\
&= (a \vee b) \wedge (a \vee c) && (5)\\
&= (a \vee b) \wedge (b \vee c) && \text{by assumption}\\
&= b \vee (a \wedge c) && (2), (5)\\
&= b \vee (b \wedge c) && \text{by assumption}\\
&= b && (9)
\end{aligned}$$

□

(11) *Uniqueness of Complements.* For all $a, b \in B$ we have

$$a \wedge b = 0 \text{ and } a \vee b = 1 \text{ imply } b = \neg a.$$

Then by setting $a = \neg b$ we obtain $b = \neg\neg b$.

Proof. To begin, we assume that $a \wedge b = 0$ and $a \vee b = 1$. Then we have

$$\begin{aligned}
a \wedge b &= 0 && \text{by assumption}\\
&= a \wedge \neg a && (4)
\end{aligned}$$

and

$$\begin{aligned}
a \vee b &= 1 && \text{by assumption}\\
&= a \vee \neg a && (4)
\end{aligned}$$

It follows from (10) that $b = \neg a$. Finally, since (4) tell us that $\neg b \wedge b = 0$ and $\neg b \vee b = 1$, we conclude that $b = \neg\neg b$. □

(12) *De Morgan's Laws.* For all $a, b \in B$ we have

$$\neg(a \vee b) = \neg a \wedge \neg b$$

$$\neg(a \wedge b) = \neg a \vee \neg b$$

Proof. By the Duality Principle we only need to prove the first statement $\neg(a \vee b) = \neg a \wedge \neg b$. By (11) it is enough to show that $(a \vee b) \wedge (\neg a \wedge \neg b) = 0$ and $(a \vee b) \vee (\neg a \wedge \neg b) = 1$. To establish these two equations, note that

$$(a \vee b) \wedge (\neg a \wedge \neg b) = [(\neg a \wedge \neg b) \wedge a] \vee [(\neg a \wedge \neg b) \wedge b] \qquad (2), (5)$$

$$
\begin{aligned}
&= [\neg b \wedge (a \wedge \neg a)] \vee [\neg a \wedge (b \wedge \neg b)] && (1), (2)\\
&= (\neg b \wedge 0) \vee (\neg a \wedge 0) && (4)\\
&= 0 \vee 0 && (8)\\
&= 0 && (3) \text{ or } (6)
\end{aligned}
$$

and

$$
\begin{aligned}
(a \vee b) \vee (\neg a \wedge \neg b) &= [(a \vee b) \vee \neg a] \wedge [(a \vee b) \vee \neg b] && (5)\\
&= [(a \vee \neg a) \vee b] \wedge [(b \vee \neg b) \vee a] && (1), (2)\\
&= (1 \vee b) \wedge (1 \vee a) && (4)\\
&= 1 \wedge 1 && (2), (8)\\
&= 1 && (3) \text{ or } (6)
\end{aligned}
$$

□

That may have seemed extremely technical, but even this proof is still expressed in a language that is readable by humans. If we don't care about readability, then the whole business of Boolean algebra can be expressed in terms of the NAND operator

$$a \uparrow b := \neg(a \wedge b) = \text{“NOT } (a \text{ AND } b)\text{”}.$$

You will prove on the homework that every Boolen function can be expressed purely in terms of NAND. What happens if we express the axioms in terms of NAND? Apparently, it is possible to define Boolean algebra with **just one axiom**:[17]

$$\boxed{((a \uparrow b) \uparrow c) \uparrow (a \uparrow ((a \uparrow c) \uparrow a)) = c.}$$

2.6 Worked Exercises

2.1. Use a truth table to verify de Morgan's laws:

$$\neg(P \wedge Q) = \neg P \vee \neg Q \qquad \text{and} \qquad \neg(P \vee Q) = \neg P \wedge \neg Q.$$

Solution. Observe that the 4th and 7th columns in each truth table are are equal:

P	Q	$P \vee Q$	$\neg(P \vee Q)$	$\neg P$	$\neg Q$	$\neg P \wedge \neg Q$
T	T	T	F	F	F	F
T	F	T	F	F	T	F
F	T	T	F	T	F	F
F	F	F	T	T	T	T

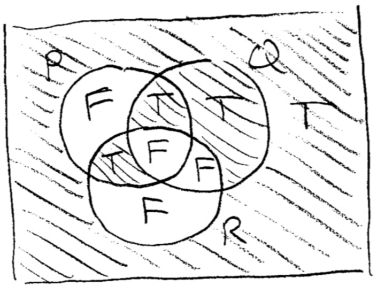
[17] This is called *Wolfram's axiom.*

P	Q	$P \wedge Q$	$\neg(P \wedge Q)$	$\neg P$	$\neg Q$	$\neg P \vee \neg Q$
T	T	T	F	F	F	F
T	F	F	T	F	T	T
F	T	F	T	T	F	T
F	F	F	T	T	T	T

2.2. Compute the disjunctive normal form of the following Boolean function. Use this to draw a circuit diagram for the function.

P	Q	R	$f(P,Q,R)$
T	T	T	F
T	T	F	T
T	F	T	T
T	F	F	F
F	T	T	F
F	T	F	T
F	F	T	F
F	F	F	T

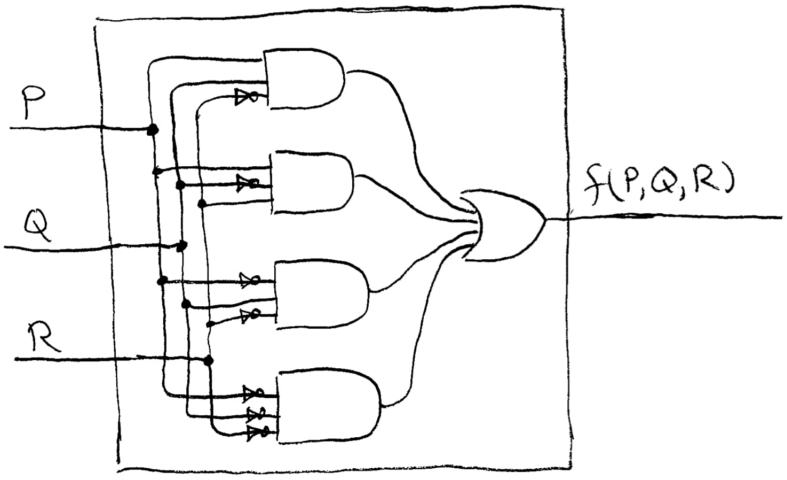
Solution. We can view the truth table as a Venn diagram:



Note that there are four shaded regions corresponding to T. We name each region using $\wedge, \neg$ and then we combine them using $\vee$:

$$f(P,Q,R) = (P \wedge Q \wedge \neg R) \vee (P \wedge \neg Q \wedge R) \vee (\neg P \wedge Q \wedge \neg R) \vee (\neg P \wedge \neg Q \wedge \neg R).$$

Here is a picture of the corresponding logic circuit:



Maybe we could simplify this circuit, but not very much.

2.3. Let B be a Boolean algebra. For all $P, Q \in B$ we define the *Sheffer stroke* as follows:

$$P \uparrow Q := \neg(P \wedge Q).$$

Use abstract Boolean algebra to prove the following identities. Don't use truth tables!

(a) $\neg P = P \uparrow P$

(b) $P \vee Q = (P \uparrow P) \uparrow (Q \uparrow Q)$

(c) $P \wedge Q = (P \uparrow Q) \uparrow (P \uparrow Q)$

In logic the Sheffer stroke is called NAND. Since any circuit can be built from OR, AND, NOT gates (by the disjunctive normal form) it follows that any circuit can be built entirely from NAND gates. This is how solid state drives work.

Solution. (a) From property (6) of Boolean algebras we have $P \wedge P = P$. Hence

$$P \uparrow P = \neg(P \wedge P) = \neg P.$$

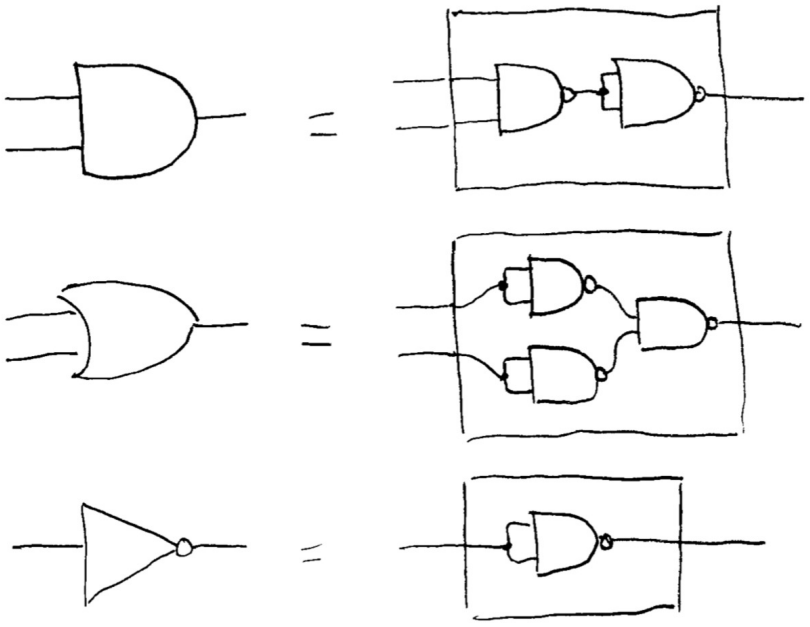
(b) By combining (a), (11) and (12) we obtain

$$\begin{aligned}(P \uparrow P) \uparrow (Q \uparrow Q) &= (\neg P) \uparrow (\neg Q) \\ &= \neg(\neg P \wedge \neg Q) \\ &= \neg\neg P \vee \neg\neg Q \\ &= P \vee Q.\end{aligned}$$

(c) By combining (a) and (11) we obtain

$$(P \uparrow Q) \uparrow (P \uparrow Q) = \neg(P \uparrow Q) = \neg(\neg(P \wedge Q)) = P \wedge Q.$$

Application: Here are the standard logic gates in terms of NAND gates:



2.?. Given $x, y, z \in \{0, 1\}$ let $b_1, b_0 \in \{0, 1\}$ be defined by $x + y + z = b_1 + 2b_0$. Use truth tables or Boolean algebra to show that

$$b_0 = x \oplus y \oplus z \qquad \text{and} \qquad b_1 = (x \wedge y) \vee (x \wedge z) \vee (y \wedge z).$$

Use these formulas to draw a simpler circuit to compute the sum of three bits.

2.4. Let $f : S \to T$ be a function of finite sets and for all $t \in T$ define the number

$$d(t) := \#\{s \in S : f(s) = t\}.$$

We say that f is *injective* if $d(t) \leqslant 1$ for all $t \in T$, *surjective* if $d(t) \geqslant 1$ for all $t \in T$ and *bijective* if $d(t) = 1$ for all T.

(a) If $f : S \to T$ is injective prove that $\#S \leqslant \#T$.

(b) If $f : S \to T$ is subjective prove tha $\#S \geqslant \#T$.

(c) If $f : S \to T$ is bijective prove that $\#S = \#T$.

[Hint: Observe that $\sum_{t\in T} d(t) = \#S$ because $\sum_{t\in T} d(t)$ is the total number of arrows and a function has exactly one arrow for each element of the domain S.]

(a) If $f : S \to T$ is injective then since $d(t) \leqslant 1$ for all $t \in T$ we have

$$\#S = \sum_{t\in T} d(t) \leqslant \sum_{t\in T} 1 = \#T.$$

(b) If $f : S \to T$ is surjective then since $d(t) \geqslant 1$ for all $t \in T$ we have

$$\#S = \sum_{t\in T} d(t) \geqslant \sum_{t\in T} 1 = \#T.$$

(a) If $f : S \to T$ is bijective then since $d(t) = 1$ for all $t \in T$ we have

$$\#S = \sum_{t\in T} d(t) = \sum_{t\in T} 1 = \#T.$$

Alternatively, since f is both injective and surjective we have $\#S \leqslant \#T$ and $\#S \geqslant \#T$, hence $\#S = \#T$.

2.5. Let S and T be finite sets. Explain why there are $\#T^{\#S}$ different functions from S to T.

To define a function we need to specify an element $f(s) \in T$ for each element $s \in S$. Since there are $\#T$ possible choices for each $f(s)$ and since these choices are completely arbitrary, the total number of choices is

$$\underbrace{\#T \times \#T \times \cdots \times \#T}_{\#S \text{ times}} = (\#T)^{\#S}.$$

2.6. This problem is about counting subsets.

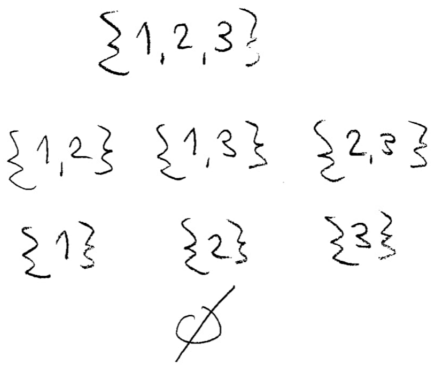
(a) Explicitly write down all of the subsets of $\{1, 2, 3\}$.

(b) Explicitly write down all of the functions $\{1, 2, 3\} \to \{T, F\}$.

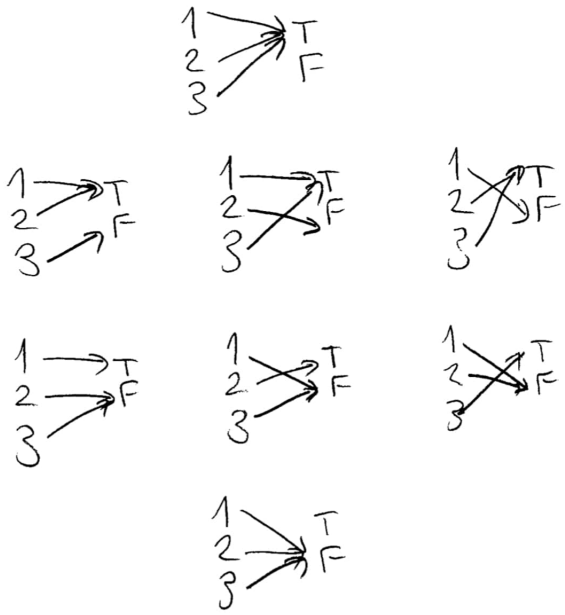
(c) For any finite set S describe a bijection between the subsets of S and the functions from $S \to \{T, F\}$.

(d) Combine Exercises 2.4(c), 2.5 and 2.6(c) to count the subsets of S.

(a) Here are the subsets:



(b) Here are the functions:



(c) Note that there is a bijection (one-to-one correspondence) between the subsets (a) and the functions (b). More generally, let S be any finite set. Then for any subset $A \subseteq S$ we define a function $f_A : S \to \{T, F\}$ as follows:

$$f_A(s) := \begin{cases} T & \text{if } s \in A, \\ F & \text{if } s \notin A. \end{cases}$$

Conversely, for any function $f : S \to \{T, F\}$ we define a subset $S_f \subseteq S$ as follows:

$$S_f := \{s \in S : f(s) = T\}.$$

Since the functions $A \mapsto f_A$ and $f \mapsto S_f$ are inverses we obtain a bijection

$$\{\text{subsets of } S\} \longleftrightarrow \{\text{functions } S \to \{T, F\}\}.$$

(d) Finally, by combining 2.4(c), 2.5 and 2.6(c) we obtain

$$\#\{\text{subsets of } S\} = \#\{\text{functions } S \to \{T, F\}\} = (\#\{T, F\})^{\#S} = 2^{\#S}.$$

Intuition: A subset of S is just a sequence of binary choices. For each of the $\#S$ elements we need to decide if it is "in" or "out". The total number of choices is

$$\underbrace{2 \times 2 \times \cdots \times 2}_{\#S \text{ times}} = 2^{\#S}.$$

2.7. This problem is about bijections from a set to itself. These are called *permutations*.

(a) How many functions are there from $\{1, 2, 3\}$ to $\{1, 2, 3\}$? (Don't write them down.)

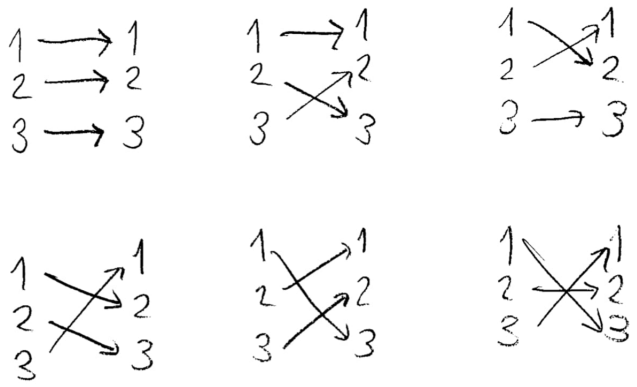
(b) How many of the functions from part (a) are bijections? Write them all down.

(c) If S is a set of size n, tell me the number of bijections $S \to S$.

(a) From Exercise 2.5 we know that the number of functions $\{1, 2, 3\} \to \{1, 2, 3\}$ is

$$(\#\{1, 2, 3\})^{(\#\{1,2,3\})} = 3^3 = 27.$$

(b) The number of bijections $\{1, 2, 3\} \to \{1, 2, 3\}$ is 6. Here they are:



(c) If $\#S = n$ then the number of bijections $S \to S$ is $n!$. Indeed, let $S = \{1, 2, \ldots, n\}$. If $f : S \to S$ is a bijection then we have n ways to choose the number $f(1)$. Say $f(1) = i$. Then

since f is injective we must have $f(2) \neq i$. Thus there are only $n-1$ ways to choose $f(2)$. Say $f(2) = j \neq i$. Again, since f is injective we must have $f(3) = k \notin \{i, j\}$. Thus there are only $n-2$ ways to choose $f(3)$. Continuing in this way, we find that the total number of choices is

$$\underbrace{n}_{\text{1st choice}} \times \underbrace{n-1}_{\text{2nd choice}} \times \underbrace{n-2}_{\text{3rd choice}} \times \cdots \times \underbrace{1}_{n\text{th choice}} = n!.$$

3 Arithmetic

In the previous chapter we discussed the foundations of logic and their application to the design of computers. But what will you do with a computer? In this chapter we will discuss the concept of "numbers" and I will present some the basic algorithms of arithmetic. The principle of induction will again play a central role.

This chapter also contains a short introduction to "number theory", which used to be called "higher arithmetic". This is the study of prime numbers and divisibility. Since the early 1600s, number theory was regarded as recreational mathematics. Since the 1970s, it plays a central role in public key cryptography. I will present the most famous cryptosystem (the RSA cryptosystem) and the mathematical theorem that makes it work (Fermat's Little Theorem).

3.1 The Integers

We all have an intuitive understanding of numbers, but in this section I will give a formal definition. To be specfic, I will give you a list of axioms for the so-called "integers":

$$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}.$$

These axioms emerged in the 1880s in the work of Richard Dedekind[18] and Giuseppe Peano.[19] (Note that this was a few decades after George Boole's work on symbolic logic.) Most of the following rules are obvious, but pay close attention to axiom (6), called the *well-ordering principle*. This axiom is logically equivalent to the principle of induction and we will discuss it further in the next section.

Definition of Integers

The system of *integers* $(\mathbb{Z}, =, <, +, \times, 0, 1)$ consists of

a set $\mathbb{Z}$,

an equivalence relation " $=$ " $\subseteq \mathbb{Z}^2$,[20]

an order relation " $<$ " $\subseteq \mathbb{Z}^2$,

two binary operations $+, \times : \mathbb{Z}^2 \to \mathbb{Z}$ called *addition* and *multiplication*,

[18] Dedekind, *Was sind und was sollen die Zahlen* [What are numbers and what should they be?] (1888).

[19] Peano, *Arithmetices principia, nova methodo exposita* [The principles of arithmetic presented by a new method] (1889).

two special elements $0, 1 \in \mathbb{Z}$ called *zero* and *one*,

which satisfy the following rules:

(1) *Definition of* $=$ *and* $<$. For all $a, b, c \in \mathbb{Z}$ we have

$a = a$

$a = b$ implies $b = a$

$a = b$ implies $a + c = b + c$ and $ac = bc$

$a = b$ and $b = c$ imply $a = c$

$a < b$ and $b < c$ imply $a < c$

exactly one of $a < b$, $a = b$ or $b < a$ is true[21]

(2) *Properties of Addition.* For all $a, b, c \in \mathbb{Z}$ we have

$a + 0 = a$

$a + b = b + a$

$a + (b + c) = (a + b) + c$

there exists "$-a$" $\in \mathbb{Z}$ such that $a + (-a) = 0$

(3) *Properties of Multiplication.* For all $a, b, c \in \mathbb{Z}$ we have

$1a = a$

$ab = ba$

$a(bc) = (ab)c$

(4) *The Distributive Law.* For all $a, b, c \in \mathbb{Z}$ we have

$a(b + c) = ab + ac$

(5) *Laws of Order.* For all $a, b, c \in \mathbb{Z}$ we have

$0 < 1$

$a < b$ implies $a + c < b + c$

if $a < b$ and $0 < c$ then $ac < bc$.

(6) *The Well-Ordering Principle.* Any non-empty set of integers that is bounded below has a least element. To be precise: We say that $S \subseteq \mathbb{Z}$ is *bounded below* if $\exists b \in \mathbb{Z}$ satisfying $b \leqslant a$ for all $a \in S$. If such b exists and if $S \neq \varnothing$ then there exists a *least element* $m \in S$ satisfying $m \leqslant a$ for all $a \in S$.[22]

You may feel that some important properties are missing, but I claim that every true fact about whole numbers is implied by these axioms. To show you how this works, I will develop the theory up to the point of "multiplicative cancellation".

Throughout the proofs I will use the property (1) without comment. I will also often use the commutative laws $a + b = b + a$ and $ab = ba$ without comment.

(7) *Uniqueness of Negatives.* For all $a \in \mathbb{Z}$ the element $-a$ is unique. In other words, if $b \in \mathbb{Z}$ satisfies $a + b = 0$ then $b = -a$. It follows from this that $-(-b) = b$.

Proof. Suppose that we have $a + b = 0$. Then I claim that $b = -a$. Indeed,

$$\begin{aligned} a + b &= a + (-a) \\ (-a) + (a + b) &= (-a) + (a + (-a)) \\ ((-a) + a) + b &= ((-a) + a) + (-a) && (2) \\ 0 + b &= 0 + (-a) && (2) \\ b &= -a && (2) \end{aligned}$$

For the second statement we put $a := -b$. Then $a + b = 0$ implies $b = -a = -(-b)$. □

Property (7) allows us to define *subtraction*: for all $a, b \in \mathbb{Z}$ we set

$$\text{“}a - b\text{”} := a + (-b).$$

(8) *Additive Cancellation.* For all $a, b, c \in \mathbb{Z}$ we have

$$a + c = b + c \text{ implies } a = b.$$

Proof. If $a + c = b + c$ then we have

$$\begin{aligned} a + c &= b + c \\ (a + c) - c &= (b + c) - c \\ a + (c - c) &= b + (c - c) && (2) \\ a + 0 &= b + 0 && (2) \\ a &= b && (2) \end{aligned}$$

[20] A general relation R on $\mathbb{Z}$ is a subset $R \subseteq \mathbb{Z} \times \mathbb{Z}$. We will use that notation aRb to indicate that $(a, b) \in R$.

[21] This is called the *law of trichotomy*.

[22] Some authors say *smallest element* instead of *least element*, but I think this might cause confusion. For example, the set $\{-2, -1, 3, 5\}$ has least element -2, whereas you might say that -1 is the **smallest element** because its absolute value is smallest. For me, "least" means "furthest to the left" on the number line.

□

(9) *Multiplication by Zero.* For all $a \in \mathbb{Z}$ we have $0a = 0$.

Proof. We have

$$\begin{aligned} 0 + 0 &= 0 && (2)\\ a(0 + 0) &= a0 \\ a0 + a0 &= a0 && (4)\\ a0 + a0 &= a0 + 0 && (2)\\ a0 &= 0 && (8) \end{aligned}$$

□

(10) *Multiplication and Negation.* For all $a, b \in \mathbb{Z}$ we have

$a(-b) = (-a)b = -(ab)$

$(-a)(-b) = ab$

Proof. To prove that $a(-b) = -(ab)$ we observe that

$$\begin{aligned} b + (-b) &= 0 && (2)\\ a(b + (-b)) &= a0 \\ a(b + (-b)) &= 0 && (9)\\ ab + a(-b) &= 0 && (4)\\ a(-b) &= -(ab) && (7) \end{aligned}$$

Then $(-a)b = -(ab)$ follows by reversing the roles of a and b. For the second statement we apply the first statement twice and then (7) to obtain

$$(-a)(-b) = -(a(-b)) = -(-(ab)) = ab.$$

So far we have not mentioned the order relation "$<$". Let's do that now.

(11) *Order and Negation.* For all $a, b, c \in \mathbb{Z}$ we have

$a < b$ implies $-b < -a$

$a < b$ and $c < 0$ implies $bc < ac$

Proof. For the first statement, assume that $a < b$. Then we have

$$\begin{aligned} a &< b \\ a + (-a) &< b + (-a) && (5) \\ 0 &< b - a \end{aligned}$$

But then $-a < -b$ is **impossible** because it would imply

$$\begin{aligned} -a &< -b \\ -a + b &< -b + b && (5) \\ b - a &< 0 \end{aligned}$$

Similarly, $-a = -b$ is **impossible** because it would imply

$$\begin{aligned} -a &= -b \\ -(-a) &= -(-b) && (7) \\ a &= b && (7) \end{aligned}$$

Since $-a < -b$ and $-a = -b$ are both false we conclude that $-b < -a$. For the second statement assume that $a < b$ and $c < 0$. Then since $-0 = 0$ we have $0 = -0 < -c$ and hence

$$\begin{aligned} a &< b \\ a(-c) &< b(-c) && (5) \\ -(ac) &< -(bc) && (10) \\ -(-(bc)) &< -(-(ac)) \\ bc &< ac && (7) \end{aligned}$$

□

(12) *Product of Nonzero Integers is Nonzero.* For all $a, b \in \mathbb{Z}$ we have

$$a \neq 0 \text{ and } b \neq 0 \text{ imply } ab \neq 0.$$

Proof. Let $a \neq 0$ and $b \neq 0$. There are four cases:

- If $0 < a$ and $0 < b$ then (5) and (9) imply $0 = 0b < ab$, hence $ab \neq 0$.
- If $0 < a$ and $b < 0$ then (5) and (9) imply $ba < 0a = a$, hence $ab \neq 0$.
- If $a < 0$ and $0 < b$ then (5) and (9) imply $ab < 0b = 0$, hence $ab \neq 0$.
- If $a < 0$ and $b < 0$ then (5) and (11) imply $0 = 0b < ab$, hence $ab \neq 0$.

□

Finally, we come to the desired result.

(13) *Multplicative Cancellation.* For all $a, b, c \in \mathbb{Z}$ we have

$$ac = bc \text{ and } c \neq 0 \text{ imply } a = b.$$

Proof. If $ac = bc$ and $c \neq 0$ then we have

$$\begin{aligned} ac &= bc \\ ac - bc &= bc - bc \\ (a-b)c &= 0 && (2), (4) \\ a - b &= 0 && (12) \\ (a-b) + b &= 0 + b \\ a &= b && (2) \end{aligned}$$

□

Remark: You may think that multiplicative cancellation is easy; if $ac = bc$ and $c \neq 0$ then just divide both sides by c to get $a = b$. The problem with this argument is that the integers don't come with a division operation. For example, it is not always possible to divide by 2 since $n/2$ is not always an integer. Below we will discuss the more general concept of "division with remainder".

In the future you may quote any of these properties without proof.

3.2 The Well-Ordering Principle

Observe that we did not use property (6) in any of the proofs of the last section. So why is this rule even necessary? Because there exist alternative number systems that satisfy all of the rules (1)–(5) but do not satisfy the well-ordering principle. For example, the system of *rational numbers*:

$$(\mathbb{Q}, =, <, +, \times, 0, 1).$$

I didn't give a formal definition of this system, but you can assume for now that it satisfies properties (1) through (5). However, I claim that $\mathbb{Q}$ does not satisfy the well-ordering principle.

Proof. Consider the set of *positive rational numbers*:

$$S = \left\{ \frac{a}{b} : a, b \in \mathbb{Z}, a \geqslant 0, b > 0 \right\} \subseteq \mathbb{Q}.$$

This set is not empty (because $1/1 \in S$) but it does not have a least element. Indeed, if you think that $\varepsilon \in S$ is the least element then you are wrong because

$$0 < \frac{\varepsilon}{2} < \varepsilon.$$

□

The reason that well-ordering failed here is because we were able squeeze an extra fraction in between 0 and ε. The next theorem shows that this does not happen for integers.

There are Gaps Between Integers

There do not exist any integers between 0 and 1. More generally, there do not exist any integers $a, b \in \mathbb{Z}$ such that

$$a < b < a + 1.$$

Proof. Let $S = \{n \in \mathbb{Z} : n > 0\}$ be the set of positive natural numbers. Since this set is not empty (indeed, we have $1 \in S$) the well-ordering principle says that there exists a *least positive integer* $m \in S$. I claim that $m = 1$.

In order to prove this, we will **assume for contradiction** that $m \neq 1$. Then from trichotomy we must have $m < 1$ or $1 < m$. But $1 < m$ is impossible because m is the least positive integer. Therefore we must have $m < 1$. Now multiply each of the inequalities $0 < m$ and $m < 1$ by the positive number m to obtain

$$\begin{array}{rcl} 0 & < & m \\ 0m & < & m^2 \\ 0 & < & m^2 \end{array} \qquad \text{and} \qquad \begin{array}{rcl} m & < & 1 \\ m^2 & < & 1m \\ m^2 & < & m. \end{array}$$

We conclude that m^2 is a positive integer that is to the left of m. Contradiction. Thus we have shown that $m = 1$ is the least positive integer. In other words, there are no integers between 0 and 1.

Next assume for contradiction that there exist integers $a, b \in \mathbb{Z}$ satisfying $a < b < a + 1$. By subtracting a from all three expressions we obtain

$$\begin{array}{rcccl} a & < & b & < & a+1 \\ 0 & < & b-a & < & 1, \end{array}$$

which contradicts our previous result. □

You might be surprised that this basic fact does not follow from the properties (1)–(5). It was the essential insight of Dedekind and Peano in the 1880s that the well-ordering principle

(or its cousin the principle of induction) is an essential property of "whole numbers". The following is another important consequence of well-ordering that is crucial to the analysis of algorithms. We will use it in the next section.

A Decreasing and Bounded Sequence of Integers Must Stop

There does not exist an infinite decreasing sequence of integers that is bounded below. In other words, suppose that the sequence $r_0, r_1, r_2, \ldots \in \mathbb{Z}$ satisfies $r_i \geqslant b$ for all i. Then it is impossible to have $r_i > r_{i+1}$ for all i.

Proof. Assume for contradiction that we have $r_i > r_{i+1} \geqslant b$ for all $i \geqslant 0$. Now define the set

$$S = \{r_0, r_1, r_2, \ldots\} \subseteq \mathbb{Z}.$$

Since this set is nonempty and bounded below by b, the well-ordering principle tells us that there exists a least element $m \in S$. By definition this element must have the form $m = r_k$ for some k. But then we have $m = r_k > r_{k+1} \geqslant b$, which implies that r_{k+1} is a **smaller** element of S. Contradiction. ▫

Similarly, one can prove that an **increasing** sequence of integers that is bounded **above** must stop. To end this section, let me officially state the relationship between well-ordering and induction.

Three Versions of Induction

The following three statements are logically equivalent.

Induction. Let $P(n)$ be a statement depending on $n \in \mathbb{Z}$. If $P(b)$ is true and if

$$\forall n \geqslant b, P(n) \Rightarrow P(n+1)$$

then $P(n)$ is true for all $n \geqslant b$.

Strong Induction. Let $P(n)$ be a statement depending on $n \in \mathbb{Z}$. If $P(b)$ is true and if

$$\forall n \geqslant b, [P(b) \wedge P(b+1) \wedge \cdots \wedge P(n)] \Rightarrow P(n+1)$$

then $P(n)$ is true for all $n \geqslant b$.

Well-Ordering. Let $S \subseteq \mathbb{Z}$ be a set of integers. If $S \neq \varnothing$ and if there exists $b \in \mathbb{Z}$ with $b \leqslant a$ for all $a \in S$ then there exists $m \in S$ with $m \leqslant a$ for all $a \in S$.

You definitely would not like to see a proof of this. Instead, see Exercise 3.2 below for an example comparing induction and well-ordering. We will see more examples in the chapter on graph theory.

3.3 The Division Algorithm

In this section we will discuss our first official algorithm and we will prove that it works.

Division With Remainder

Given integers $n \geqslant 0$ and $d > 0$ we wish to find integers $q, r \in \mathbb{Z}$ satisfying

$$\begin{cases} n = qd + r, \\ 0 \leqslant r < d. \end{cases}$$

I claim that the following algorithm works.

procedure: divide n by d with remainder
input: (n, d)
initialize: $(q, r) := (0, n)$
while $r \geqslant d$ **do**
    $q := q + 1$
    $r := r - d$
output: (q, r)

Furthermore, I claim that the resulting quotient and remainder are unique.

Proof that the algorithm works. Set $q := 0$ and $r := n$ and observe that $n = qd + r$. If $n < d$ then this is the correct answer. Otherwise, we enter the while loop. On each iteration the equation $n = dq + r$ is preserved. Indeed, if $n = dq + r$ then we also have

$$(q + 1)d + (r - d) = qd + r = n.$$

If the while loop terminates, then we must have $r \ngeqslant d$ and hence $r < d$. Furthermore, we must have $0 \leqslant r$ because on the previous iteration we had $r \geqslant d$ and hence $r - d \geqslant 0$. Finally, I claim that the while loop does indeed terminate. To see this, let us assume for contradiction

that the algorithm goes on forever. Then since $d > 0$ we obtain an infinite decreasing sequence of integers that is bounded below by d:

$$r > r - d > r - d - d > r - d - d - d > \cdots \geqslant d.$$

This violates the well-ordering principle. □

Proof of uniqueness. Suppose that we each write our own algorithm to compute the quotient and remainder of (n, d). Suppose that I run my algorithm and get (q_1, r_1), while you run your algorithm and get (q_2, r_2). Then I claim that we must have $q_1 = q_2$ and $r_1 = r_2$. To see this, we first reiterate the definition of quotient and remainder:

$$\begin{cases} n = q_1 d + r_1, \\ 0 \leqslant r_1 < d, \end{cases} \qquad \text{and} \qquad \begin{cases} n = q_2 d + r_2, \\ 0 \leqslant r_2 < d. \end{cases}$$

In particular, we must have $q_1 d + r_1 = n = q_2 d + r_2$ and hence

$$\begin{aligned} q_1 d + r_1 &= q_2 d + r_2 \\ (q_1 - q_2)d &= (r_2 - r_1). \end{aligned}$$

Our goal is to show that $r_1 = r_2$, so let us assume for contradiction that $r_2 - r_1 \neq 0$. By the law of trichotomy this implies that $r_2 - r_1 > 0$ or $r_1 - r_2 < 0$. We will only treat the first case and leave the other case to the reader. So let us assume that $r_2 - r_1 > 0$. Then since $(q_1 - q_2)d = (r_2 - r_1)$ and $d > 0$ we must also have $q_1 - q_2 > 0$. Since $q_1 - q_2$ is a whole number this implies that[23]

$$\begin{aligned} 1 &\leqslant q_1 - q_2 \\ d &\leqslant (q_1 - q_2)d \\ d &\leqslant r_2 - r_1. \end{aligned}$$

On the other hand, since $0 \leqslant r_1$ and $r_2 < d$ we have $r_2 - r_1 < d - r_1 \leqslant d$. Combining these inequalities gives

$$d \leqslant r_2 - r_1 < d,$$

which is a contradiction. Using a similar argument, the assumption $r_2 - r_1 < 0$ also leads to a contradiction. Therefore we conclude that $r_2 - r_1 = 0$ and hence $r_1 = r_2$. Finally, since $d(q_1 - q_2) = (r_2 - r_1) = 0$ and $d \neq 0$ we conclude by cancellation that $q_1 - q_2 = 0$ and hence $q_1 = q_2$.[24] □

For example, suppose we want to divide $n = 31$ by $d = 7$. Here are the steps of the algorithm:

(q, r)	$r \geqslant 7$?
$(0, 31)$	yes
$(1, 24)$	yes
$(2, 17)$	yes
$(3, 10)$	yes
$(4, 3)$	no

[23] Recall, there are no integers between 0 and 1.

[24] Details: If $ad = bd$ and $d \neq 0$ then cancellation says that $a = b$. In this case we use $a = q_1 - q_2$ and $b = 0$.

We conclude that the quotient is $q = 4$ and the remainder is $r = 3$. Indeed, we observe that

$$\begin{cases} 31 = 4 \cdot 7 + 3, \\ 0 \leqslant 3 < 7, \end{cases}$$

as desired. Note that "division with remainder" is partly an algorithm and partly a theorem. We will see in the next sections that it has many applications.

3.4 Base b Arithmetic

Why do we use the following ten symbols to denote numbers?

$$0, 1, 2, 3, 4, 5, 6, 7, 8, 9$$

There is no good reason. Apparently, our common "decimal system" (also called the Hindu-Arabic numeral system) was developed in India around 600AD and spread outward from there. It was promoted in Europe by Leonardo of Pisa[25] in the *Liber Abaci* (1202). The decimal system was a huge technological advance because it comes with efficient algorithms for the basic operations of arithmetic.

Let me remind you how this works. The basic symbols $0, 1, 2, 3, 4, 5, 6, 7, 8, 9$ are defined as the first ten consecutive natural numbers. Compound symbols such as "512" represent a sum of powers of ten:

$$\text{“73512”} = 2 \cdot 10^0 + 1 \cdot 10^1 + 5 \cdot 10^2 + 3 \cdot 10^3 + 7 \cdot 10^4 + 0 \cdot 10^5 + 0 \cdot 10^6 + \cdots .$$

But there is nothing special about "base ten". The following theorem tells us that we can use any base $b \geqslant 2$ as the foundation for arithmetic. The base $b = 2$ is particularly convenient for implementing arithmetic on a computer.

Base b Expansion of Integers (Positional Notation)

Fix a base $b \geqslant 2$ and let $n \geqslant 0$ be any non-negative integer. We wish to find a sequence of numbers $r_0, r_1, r_2, \ldots, r_k$ from the set $\{0, 1, \ldots, b-1\}$ such that

$$n = r_0 + r_1 b + r_2 b^2 + r_3 b^3 + \cdots + r_k b^k.$$

In the following algorithm we will use the notation "a **rem** b" for the remainder and "a **quo** b" for the quotient when a is divided by b:[26]

procedure: base b expansion of n
input: (n, b)
initialize:
  $q := n$
  $k := 0$
while $q \neq 0$ **do**

[25] Leonardo of Pisa is also called "Fibonacci".

$r_k := q \,\mathbf{rem}\, b$
$q := q \,\mathbf{quo}\, b$
$k := k + 1$
output: $(r_k, \ldots, r_2, r_1, r_0)$

I claim that the output satisfies $n = r_0 + r_1 b + \cdots + r_k b^k$. Furthermore, I claim that the number k and the sequence $r_0, r_1, \ldots, r_k$ are unique. In this case we will write

$$n = (r_k \cdots r_2 r_1 r_0)_b.$$

Remark: We can also represent fractions and real numbers by allowing negative powers of b, but we won't pursue this here.

Proof that the algorithm works. Let me write the algorithm more explicitly. We begin by computing $q_0 := n \,\mathbf{quo}\, b$ and $r_0 := n \,\mathbf{rem}\, b$. If $q_0 \neq 0$ then we continue with $q_1 := q_0 \,\mathbf{quo}\, b$ and $r_1 := q_0 \,\mathbf{rem}\, b$. If $q_1 \neq 0$ then we continue to obtain

$$\begin{aligned}
n &= bq_0 + r_0 && 0 \leqslant r_0 < b \\
q_0 &= bq_1 + r_1 && 0 \leqslant r_1 < b \\
q_1 &= bq_2 + r_2 && 0 \leqslant r_3 < b \\
&\vdots \\
q_{k-2} &= bq_{k-1} + r_{k-1} && 0 \leqslant r_{k-1} < b \\
q_{k-1} &= b0 + r_k && 0 \leqslant r_k < b
\end{aligned}$$

If the procedure never terminates then since $b > 1$ we must have

$$q_i = bq_{i+1} + r_{i+1} > q_{i+1} + r_{i+1} \geqslant q_{i+1} \qquad \text{for all } i.$$

But then we obtain an infinite decreasing sequence of positive integers

$$q_0 > q_1 > q_2 > \cdots > 0,$$

which violates the well-ordering principle. Thus the procedure must terminate. Finally, we check that the answer is correct:

$$\begin{aligned}
n &= q_0 b + r_0 \\
&= (q_1 b + r_1) b + r_0 \\
&= q_1 b^2 + r_1 b + r_0 \\
&= (q_2 b + r_2) b^2 + r_1 b + r_0 \\
&= q_2 b^3 + r_2 b^2 + r_1 b + r_0
\end{aligned}$$

[26] Some programming languages use a **mod** b instead of $a \,\mathbf{rem}\, b$.

$$
\begin{aligned}
&\vdots \\
&= q_{k-1}b^k + r_{k-1}b^{k-1} + \cdots + r_2b^2 + r_1b + r_0 \\
&= (0b + r_k)b^k + r_{k-1}b^{k-1} + \cdots + r_2b^2 + r_1b + r_0 \\
&= r_kb^k + r_{k-1}b^{k-1} + \cdots + r_2b^2 + r_1b + r_0.
\end{aligned}
$$

□

Alternate proof of existence. Alternatively, we can prove by induction that every positive integer has a base b expansion. We begin by observe that 0 and 1 have base b expansions:

$$
\begin{aligned}
0 &= 0 + 0b + 0b^2 + \cdots, \\
1 &= 1 + 0b + 0b^2 + \cdots.
\end{aligned}
$$

Now I just have to tell you how to "add 1". For example, in base 10 we have

$$15299999 + 1 = 15300000.$$

In other words, if there is a string of 9s on the right, then we replace each 9 by 0 and add 1 to the first digit on the left. In general, suppose that we have

$$n = (b-1) + (b-1)b + (b-1)b^2 + \cdots (b-1)b^k + r_{k+1}b^{k+1} + r_{k+2}b^{k+2} + \cdots$$

with $0 \leqslant r_{k+1} < b - 1$.[27] Then I claim that $n + 1$ has the expansion

$$n + 1 = 0 + 0b + 0b^2 + \cdots + 0b^k + (r_{k+1} + 1)b^{k+1} + r_{k+2}b^{k+2} + \cdots,$$

which is valid because $r_{k+1} + 1 < b$. To see this we will use the geometric series:

$$\boxed{1 + b + b^2 + \cdots + b^k = \frac{b^{k+1} - 1}{b - 1}.}$$

Then we have

$$
\begin{aligned}
n &= (b-1) + (b-1)b + (b-1)b^2 + \cdots (b-1)b^k + r_{k+1}b^{k+1} + r_{k+2}b^{k+2} + \cdots \\
n &= (b-1)(1 + b + b^2 + \cdots + b^k) + r_{k+1}b^{k+1} + r_{k+2}b^{k+2} + \cdots \\
n &= (b-1)\frac{b^{k+1} - 1}{b - 1} + r_{k+1}b^{k+1} + r_{k+2}b^{k+2} + \cdots \\
n &= (b^{k+1} - 1) + r_{k+1}b^{k+1} + r_{k+2}b^{k+2} + \cdots \\
n + 1 &= (r_{k+1} + 1)b^k + r_{k+2}b^{k+2} + \cdots.
\end{aligned}
$$

□

[27] Such an integer k must exist by well-ordering.

Proof of uniqueness. Suppose that we have two sequences $r_0, r_1, \ldots \in \{0, 1, \ldots, b-1\}$ and $s_0, s_1, \ldots \in \{0, 1, \ldots, b-1\}$ satisfying

$$r_0 + r_1 b + r_2 b^2 + \cdots = s_0 + s_1 b + s_2 b^2 + \cdots .$$

We will prove by induction that $r_i = s_i$ for all $i \geqslant 0$. First we divide each side by b to obtain

$$r_0 + b(r_1 + r_2 b + \cdots) = s_0 + b(s_1 + s_2 b + \cdots).$$

Since $r_0, s_0 \in \{0, 1, \ldots, b-1\}$ the uniqueness of remainders implies that $r_0 = s_0$. Now fix some n and assume for induction that we have $r_i = s_i$ for all $0 \leqslant i \leqslant n$. Then we subtract the number $r_0 + \cdots r_n b^n = s_0 + \cdots s_n b^n$ from both sides and factor out b^{n+1} to obtain

$$\begin{aligned}
\cancel{r_0 + r_1 b + \cdots + r_n b^n} + r_{n+1} b^{n+1} + \cdots &= \cancel{s_0 + s_1 b + \cdots + s_n b^n} + s_{n+1} b^{n+1} + \cdots \\
r_{n+1} b^{n+1} + r_{n+2} b^{n+2} + \cdots &= s_{n+1} b^{n+1} + s_{n+2} b^{n+2} + \cdots \\
\cancel{b^{n+1}}(r_{n+1} + r_{n+2} b + \cdots) &= \cancel{b^{n+1}}(s_{n+1} + s_{n+2} b + \cdots) \\
r_{n+1} + r_{n+2} b + r_{n+3} b^2 + \cdots &= s_{n+1} + s_{n+2} b + s_{n+3} b^2 + \cdots .
\end{aligned}$$

Finally, we divide both sides by b to get

$$r_{n+1} + b(r_{n+2} + r_{n+3} b + \cdots) = s_{n+1} + b(s_{n+2} + s_{n+3} b + \cdots).$$

Since $r_{n+1}, s_{n+1} \in \{0, \ldots, b-1\}$ the uniqueness of remainders implies that $r_{n+1} = s_{n+1}$. ▫

For example, let us expand the decimal number $11 = (11)_{10}$ in the bases $b = 2, 3, 4$. We can use the slow method of "repeatedly adding 1":

base 10	base 2	base 3	base 4
0	0	0	0
1	1	1	1
2	10	2	2
3	11	10	3
4	100	11	10
5	101	12	11
6	110	20	12
7	111	21	13
8	1000	22	20
9	1001	100	21
10	1010	101	22
11	1011	102	23

We conclude that

$$\begin{aligned}
11 &= (1011)_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0, \\
11 &= (102)_3 = 1 \cdot 3^2 + 0 \cdot 3^1 + 2 \cdot 3^0,
\end{aligned}$$

$$11 = (23)_4 = 2 \cdot 4^1 + 3 \cdot 4^0.$$

Or we can use the faster method of "repeatedly dividing the quotient by the base":

$$\begin{array}{rcl} \mathbf{11} & = & \mathbf{5} \cdot 2 + 1 \\ \mathbf{5} & = & \mathbf{2} \cdot 2 + 1 \\ \mathbf{2} & = & \mathbf{1} \cdot 2 + 0 \\ \mathbf{1} & = & \mathbf{0} \cdot 2 + 1 \end{array} \qquad \begin{array}{rcl} \mathbf{11} & = & \mathbf{3} \cdot 3 + 2 \\ \mathbf{3} & = & \mathbf{1} \cdot 3 + 0 \\ \mathbf{1} & = & \mathbf{0} \cdot 3 + 1 \end{array} \qquad \begin{array}{rcl} \mathbf{11} & = & \mathbf{2} \cdot 4 + 3 \\ \mathbf{2} & = & \mathbf{0} \cdot 4 + 2 \end{array}$$

Observe that the sequences of remainders give the same answer as above.

If $b > 10$ then you will need to invent some new symbols. The common choice is to use uppercase Roman letters. In the *hexadecimal system* (base $b = 16$) we use the symbols

$$0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F$$

for the numbers zero through fifteen. Thus, for example, we have

$$\begin{aligned} (3B5)_{16} &= 3 \cdot 16^2 + B \cdot 16^1 + 5 \cdot 16^0 \\ &= 3 \cdot 256 + 11 \cdot 16 + 5 \\ &= 949. \end{aligned}$$

Positional notation gives us convenient algorithms for the basic operations of arithmetic. I'm sure you have learned these algorithms in base 10 and then probably forgot them. Examples.

3.5 The Euclidean Algorithm

The Euclidean algorithm is another important number-theoretic algorithm that has nothing to do with base b notation. It is used extensively in cryptography. The purpose of the Euclidean algorithm is to compute the greatest common divisor of two integers a, b. More generally, given any three integers $a, b, c \in \mathbb{Z}$ the Euclidean algorithm can be used to find all integer solutions $x, y \in \mathbb{Z}$ to the equation

$$ax + by = c.$$

Here is the fundamental definition on which everything else is based.

Definition of Divisibility

Given two integers $n, d \in \mathbb{Z}$ with $d > 0$ we define the statement "$d|n$" as follows:[28]

$$\text{"}d|n\text{"} \iff \text{"}\exists k \in \mathbb{Z}, dk = n\text{"} \iff (n \,\mathbf{rem}\, d) = 0$$

In words, we say that "*d divides n*" or that "*n is divisible by d*".

Remarks:

- For all $n \in \mathbb{Z}$ we have $n|n$ because $n1 = n$ $(k = 1)$.
- For all $n \in \mathbb{Z}$ we have $1|n$ because $1n = n$ $(k = n)$.
- For all $n \in \mathbb{Z}$ we have $n|0$ because $n0 = 0$ $(k = 0)$.
- Normally we don't allow $d = 0$. If we did then we would have $0 \nmid n$ except when $n = 0$ because $0k = 0$ for all $k \in \mathbb{Z}$. I guess we would also have $0|0$, but never mind.

Next, suppose we have $dk = n$ for some positive d, k, n. Then it follows that

$$\begin{aligned} 1 &\leqslant k \\ d &\leqslant dk \\ d &\leqslant n. \end{aligned}$$

In other words, the divisors of n are no bigger than n. For any two positive integers $a, b \in \mathbb{Z}$, it follows that the set of common divisors of a and b is bounded above by the minimum of a and b, hence this set has a greatest element, called the "greatest common divisor".[29]

Greatest Common Divisor

Let a and b be positive integers. Then we let $d = \gcd(a, b)$ denote the *greatest common divisor* of a and b. To be precise, this number satisfies the following two properties:

- $d|a$ and $d|b$
- if $c|a$ and $c|b$ then $c \leqslant d$.

We also define $\gcd(a, 0) = a$, since 0 is divisible by every number, so the common divisors of a and 0 are just the divisors of a, and the greatest divisor of a is just a itself.

For example, let $d = \gcd(12, 30)$, so that $1 \leqslant d \leqslant 12$. In order to compute d we can just list the numbers up to 12 and strike out the numbers that do not divide 12 and 30:

$$1, 2, 3, \cancel{4}, \cancel{5}, 6, \cancel{7}, \cancel{8}, \cancel{9}, \cancel{10}, \cancel{11}, \cancel{12}.$$

[28] We could also allow $d < 0$ but this won't come up in our applications.

[29] The original well-ordering principle says that any non-empty set of integers that is bounded below has a least element. Here I am using the equivalent statement that any non-empty set of integers that is bounded **above** has a **greatest** element.

We conclude that $d = 6$. If $a > b$ then this algorithm computes $\gcd(a, b)$ in b steps, by listing all the numbers up to b and striking out those that do not divide a or b.[30] This is unacceptably slow for modern cryptographic purposes, which routinely deal with integers with hundreds of digits. Luckily, there is an infinitely better algorithm.[31] This algorithm is truly ancient; it appears in Euclid's *Elements*, Book X, from approximately 300BC.

The Euclidean Algorithm

Given any integers $a > b \geqslant 0$, the following algorithm computes the greatest common divisor $\gcd(a, b)$.

procedure: to compute $\gcd(a, b)$
input: (a, b) with $a > b \geqslant 0$
initialize:
  $r_0 := a$
  $r_1 := b$
  $k := 1$
while $r_k > 0$ **do**
  $r_{k+1} := r_{k-1}$ **rem** r_k
  $k := k + 1$
output: r_{k-1}

The proof requires a small lemma, which you will prove on the homework.

Small Lemma

For any positive integers $a, b, c, x \in \mathbb{Z}$ satisfying $a = bx + c$ we must have

$$\gcd(a, b) = \gcd(b, c).$$

Proof that the algorithm works. Let me write the algorithm more explicitly. First we define $r_0 := a$ and $r_1 := b$. If $r_1 = 0$ then we report $\gcd(a, b) = r_0 = a$, otherwise we divide r_0 by r_1 to obtain integers $q_2, r_2 \in \mathbb{Z}$ satisfying

$$\begin{cases} r_0 = r_1 q_2 + r_2, \\ 0 \leqslant r < d. \end{cases}$$

[30] Technically, I guess it's $2b$ steps, since at each step we have to compute two divisions with remainder.

[31] If $a > b$ then Lamé's Theorem below shows that the Euclidean Algorithm computes $\gcd(a, b)$ in less than $5 \cdot \log_{10}(a)$ steps.

If $r_2 = 0$ then we report $\gcd(a, b) = r_1 = b$. Otherwise, we continue to divide r_i by r_{i+1} until we obtain $r_k = 0$:

$$\begin{aligned} r_0 &= r_1 q_2 + r_2 && 0 < r_2 < r_1 \\ r_1 &= r_2 q_3 + r_3 && 0 < r_3 < r_2 \\ &\vdots \\ r_{k-3} &= r_{k-2} q_{k-1} + r_{k-1} && 0 < r_{k-1} < r_{k-2} \\ r_{k-2} &= r_{k-1} q_k + 0. \end{aligned}$$

We know that we must eventually have $r_k = 0$ for some k, since otherwise we obtain an infinite decreasing sequence of positive remainders:

$$r_0 > r_1 > r_2 > \cdots > 0.$$

Finally, we need to check that the last nonzero remainder r_{k-1} is equal to $\gcd(a, b)$. Indeed, by repeatedly applying the Small Lemma, we have

$$\begin{aligned} \gcd(a, b) &= \gcd(r_0, r_1) \\ &= \gcd(r_1, r_2) \\ &\vdots \\ &= \gcd(r_{k-2}, r_{k-1}) \\ &= \gcd(r_{k-1}, r_k) \\ &= \gcd(r_{k-1}, 0) \\ &= r_{k-1}. \end{aligned}$$

□

To illustrate the algorithm we compute the greatest common divisor of 3094 and 2513:

$$\begin{aligned} \mathbf{3094} &= \mathbf{2513} \cdot 1 \quad +\mathbf{581} \\ \mathbf{2513} &= \mathbf{581} \cdot 4 \quad +\mathbf{189} \\ \mathbf{581} &= \mathbf{189} \cdot 3 \quad +\mathbf{14} \\ \mathbf{189} &= \mathbf{14} \cdot 13 \quad +\mathbf{7} \\ \mathbf{14} &= \mathbf{7} \cdot 2 \quad +\mathbf{0} \end{aligned}$$

Since the last nonzero remainder is 7 we conclude that $\gcd(3094, 2513) = 7$. Note that the algorithm stopped after 5 steps. This is extremely fast, since the naive method takes $2 \cdot 2513 = 5026$ steps. The following theorem proves that the running time of the Euclidean Algorithm is always less than 5 times the number of decimal digits in the larger number.[32]

[32] Gabriel Lamé, *Note sur la limite du nombre des divisions dans la recherche du plus grand commun diviseur entre deux nombres entiers* (1844).

Lamé's Theorem (1844)

Given $a > b \geqslant 0$, the number of steps used to compute $\gcd(a, b)$ is less than

$$4.785 \cdot \log_{10}(a) + 1 \ .$$

Proof.[33] The proof surprisingly uses the Fibonacci numbers. Recall that

$$F_n := \begin{cases} 0 & \text{if } n = 0, \\ 1 & \text{if } n = 1, \\ F_{n-1} + F_{n-2} & \text{if } n \geqslant 2. \end{cases}$$

We will prove by induction that the following statement holds for all $n \geqslant 0$:

> "For all integers $a > b \geqslant 0$, if the Euclidean algorithm for $\gcd(a, b)$ requires n divisions with remainder then we must have $a \geqslant F_{n+1}$ and $b \geqslant F_n$".

For the base case, let $a > b \geqslant 0$ and assume that we can compute $\gcd(a, b)$ using $n = 0$ steps. In this case we must have $b = 0 \geqslant F_0$ and $a > b = 0$, hence $a \geqslant 1 = F_1$ as desired. Now fix some $n \geqslant 0$ and assume that the statement holds for the values $0, 1, 2, \ldots, n$. In this case we will show that the statement holds for $n + 1$. So consider some $a > b \geqslant 0$ and assume that it takes $n + 1$ divisions with remainder to compute $\gcd(a, b)$. The first step of the Euclidean algorithm divides a by b to get

$$a = qb + r \qquad \text{with } 0 \leqslant r < b.$$

Then the algorithm proceeds recursively to compute $\gcd(b, r)$ in n steps. By induction this implies that $b \geqslant F_{n+1}$ and $r \geqslant F_n$. Finally, since $a > b$ we must have $q \geqslant 1$ and hence

$$a = qb + r \geqslant b + r = F_{n+1} + F_n = F_{n+2}.$$

We have proved that $a \geqslant F_{n+2}$ and $b \geqslant F_{n+1}$ as desired.

Finally, we will use the fact proved on a previous homework that $F_{n+1} > \varphi^{n-1}$ for all $n \geqslant 2$, where $\varphi = (1 + \sqrt{5})/2 = 1.61$ is the golden ratio. Let $a > b \geqslant 0$ and suppose that the computation of $\gcd(a, b)$ takes $n \geqslant 2$ steps. Then we have

$$\begin{aligned} \varphi^{n-1} &< F_{n+1} \leqslant a \\ \varphi^{n-1} &< a \\ n - 1 &< \log_{\varphi}(a) \\ n - 1 &< \log_{10}(a) / \log_{10}(\varphi) \end{aligned}$$

[33] The proof is subtle, so feel free to skip it.

$$\begin{aligned} n - 1 &< 4.785 \cdot \log_{10}(a) \\ n &< 4.785 \cdot \log_{10}(a) + 1. \end{aligned}$$

□

Sometimes the algorithm runs faster than predicted. For example, we computed $\gcd(3094, 2513)$ in 5 steps, whereas Lamé predicts $4.785 \cdot \log_{10}(3094) + 1 \approx 18$ steps. In general, large quotients make the algorithm run faster than predicted. The worst case scenario is when the quotient in each step equals 1. This occurs when we compute the gcd of two consecutive Fibonacci numbers. Recall that the Fibonacci numbers begin as follows:

n	0	1	2	3	4	5	6	7	8
F_n	0	1	1	2	3	5	8	13	21

It turns out that the computation of $\gcd(F_n, F_{n-1})$ always takes exactly $n - 1$ steps, which matches Lamé's bound.[34] For example, here is the computation of $\gcd(F_8, F_7) = \gcd(21, 8)$:[35]

$$\begin{aligned} \mathbf{21} &= 1 \cdot \mathbf{13} + \mathbf{8} \\ \mathbf{13} &= 1 \cdot \mathbf{8} + \mathbf{5} \\ \mathbf{8} &= 1 \cdot \mathbf{5} + \mathbf{3} \\ \mathbf{5} &= 1 \cdot \mathbf{3} + \mathbf{2} \\ \mathbf{3} &= 1 \cdot \mathbf{2} + \mathbf{1} \\ \mathbf{2} &= 2 \cdot \mathbf{1} + \mathbf{0} \end{aligned}$$

This special property of Fibonacci numbers is the reason for their appearance in the proof.

3.6 Introduction to Cryptography

I will end this chapter with a highly non-trivial application of arithmetic. As mentioned in the introduction, the theory of prime numbers was long regarded as recreational mathematics. However, in the 1970s it suddenly became the basis of secure electronic communication. Instead of describing the theory of cryptography in general, I will describe in detail the most important cryptosystem. This system is called RSA for Rivest, Shamir and Adelman, who published the method in 1977 and received a patent in 1983.[36]

First allow me to describe the system without explaining how it works.

[34] The answer is always 1, which is not interesting. The interesting point is the running time of the algorithm.

[35] The last step is a bit of an anomaly. We could make the pattern cleaner by writing $\mathbf{2} = 1 \cdot \mathbf{1} + \mathbf{1}$ and then $\mathbf{1} = 1 \cdot \mathbf{1} + \mathbf{0}$, even though that slightly breaks the definition of quotient and remainder.

[36] An equivalent system was developed in 1973 by Clifford Cooks, working at GCHQ (British signals intelligence), which was declassified in 1997. Poor Clifford did not get rich and his name is relegated to footnotes. For more information see *The Code Book* by Simon Singh.

RSA Cryptosystem

Alice wants to receive a secret message from Bob over an insecure channel. To set up the system Alice performs the following steps:

Setup by Alice.

- Choose two large random prime numbers p and q.[37]
- Compute $n = pq$.
- Choose a random number e satisfying $\gcd(e, (p-1)(q-1)) = 1$.
- Use the Extended Euclidean Algorithm to find integers d, k satisfying

$$de = (p-1)(q-1)k + 1.$$

- Publish the numbers (n, e) as the *public key*.
- Keep the numbers d, p, q secret.

Now Bob uses the public key (n, e) to send a message:

Encryption by Bob.

- Convert the secret message to an integer $0 \leqslant m < n$.[38]
- Compute the remainder $c = m^e \,\mathbf{rem}\, n$.
- Send the number c to Alice.

Finally, Alice uses the private key d to decrypt the message:

Decryption by Alice.

- Compute the remainder $m' = c^d \,\mathbf{rem}\, n$.
- Theorem: $m' = m$ is Bob's original message.

Suppose that Eve the eavesdropper is intercepting communications, so Eve knows the numbers n, e and c. In principle, Eve could recover m by factoring n into pq and then recreating Alice's computation to obtain d. The security of the system is based on the following assumption:

> By choosing the primes p and q to be sufficiently large, it can be arranged that Alice's and Bob's computations to set up and use the system are arbitrarily cheaper than the computations necessary for Eve to factor n into pq.

No one has ever proved a theorem to this effect, but after decades of use it seems that RSA is secure.[39]

It will take the rest of the section to describe how to perform the computations and to prove that $m' = m$. Even then we will skip some steps. First we will describe how Alice computes d and k. Recall that for any integers $a > b \geqslant 0$, the Euclidean algorithm computes $\gcd(a, b)$. By slightly modifying the algorithm we can also find integers $x, y \in \mathbb{Z}$ such that

$$ax + by = \gcd(a, b).$$

Before giving the formal statement I will explain the intuition behind it. The idea is to consider triples of integers (x, y, r) satisfying $ax + by = r$. There are two obvious such triples: $(1, 0, a)$ and $(0, 1, b)$. Furthermore, if we have two triples (x, y, r) and (x', y', r') satisfying $ax + by = r$ and $ax' + by' = r'$ then for any integer q, the triple

$$(*) \qquad (x'', y'', r'') := (x, y, r) - q(x', y', r') = (x - qx', y - qy', r - qr')$$

also satisfies

$$\begin{aligned} ax'' + by'' &= a(x - qx') + b(y - qy') \\ &= (ax + by) - q(ax' + by') \\ &= r - qr' \\ &= r''. \end{aligned}$$

Beginning with the obvious triples $(1, 0, a)$ and $(0, 1, b)$, the goal is to repeatedly combine these triples using $(*)$ until we obtain a triple of the form $(x, y, \gcd(a, b))$, and the steps of the Euclidean Algorithm tell us exactly how to do this.

For example, let's take $a = 3094$ and $b = 2513$. In the previous section we found that $\gcd(3094, 2513) = 7$. Now we will find integers $x, y \in \mathbb{Z}$ satisfying $3094x + 2513y = 7$. The following table shows the steps of the computation:

x	y	r
1	0	3094
0	1	2513
1	-1	581
-4	5	189
13	-16	14
-173	213	7
359	-442	0

Let's name the first two rows:

$$(x_0, y_0, r_0) = (1, 0, 3094),$$

[37] Mnemonic: p is for prime, n is for number, e is for encryption, d is for decryption, m is for message, c is for ciphertext.

[38] There are many standard ways to do this; for example, ASCII.

[39] If and when quantum computers become feasible then RSA will be broken by Shor's factorization algorithm.

$$(x_1, y_1, r_1) = (0, 1, 2513).$$

To obtain the next row we take

$$\begin{aligned}(x_2, y_2, r_2) &= (x_0, y_0, r_0) - 1(x_1, y_1, r_1) \\ &= (1, 0, 3094) - 1(0, 1, 2513) \\ &= (1, -1, 581).\end{aligned}$$

More generally, we define

$$(x_{n+1}, y_{n+1}, r_{n+1}) = (x_{n-1}, y_{n-1}, r_{n-1}) - q_{n+1}(x_n, y_n, r_n),$$

there the q_n are the quotients produced by the Euclidean Algorithm. This guarantees that we will eventually reach a triple of the form $(x, y, 7)$, so that

$$3094x + 2513y = 7.$$

From the above table we find that[40]

$$3094(-173) + 2513(213) = 7.$$

Let me remark that the solution (x, y) is **not unique**. Indeed, the final row in the table tells us that

$$3094(359) + 2513(-442) = 0,$$

and we can multiply this by any integer ℓ to get

$$3094(359\ell) + 2513(-442\ell) = 0.$$

Finally, combining the two equations gives

$$\begin{aligned}7 &= 7 + 0 \\ &= 3094(-173) + 2513(213) + 3094(359\ell) + 2513(-442\ell) \\ &= 3094(-173 + 359\ell) + 2513(213 - 442\ell),\end{aligned}$$

so that $(x, y) = (-173 + 359\ell, 213 - 442\ell)$ is a solution for any $\ell \in \mathbb{Z}$.[41]

Here is the official statement and its proof.

The Extended Euclidean Algorithm

Given integers $a > b \geqslant 0$, the following modified version of the Euclidean Algorithm will find integers $x, y \in \mathbb{Z}$ satisfying

$$ax + by = \gcd(a, b).$$

procedure: to find $x, y \in \mathbb{Z}$ such that $ax + by = \gcd(a, b)$

[40] This would be very difficult to find by trial and error!

[41] In fact this formula gives **all** solutions to the equation $3094x + 2513y = 7$, but we won't prove it.

input: (a, b) with $a > b \geqslant 0$
initialize:
$r_0 := a$
$r_1 := b$
$k := 1$
$(x_0, y_0) = (1, 0)$
$(x_1, y_1) = (0, 1)$
while $r_k > 0$ **do**
$r_{k+1} := r_{k-1}$ **rem** r_k
$q_{k+1} := r_{k-1}$ **quo** r_k
$(x_{k+1}, y_{k+1}) := (x_k, y_k) - q_{k+1}(x_{k-1}, y_{k-1})$
$k := k + 1$
output: (x_{k-1}, y_{k-1})

Proof that the algorithm works. Note that we have merely added more variables to the usual Euclidean Algorithm, so the remainders still satisfy the property that $r_k = 0$ and $r_{k-1} = \gcd(a, b)$. To prove the result we will show by (strong) induction that

$$(*) \qquad ax_n + by_n = r_n \quad \text{for all } 0 \leqslant n \leqslant k.$$

Then it will follow that the output (x_{k-1}, y_{k-1}) satisfies

$$ax_{k-1} + by_{k-1} = r_{k-1} = \gcd(a, b),$$

as desired. For the base cases we observe that

$$\begin{aligned} ax_0 + by_0 &= a1 + b0 = a = r_0, \\ ax_1 + by_1 &= a0 + b1 = b = r_1. \end{aligned}$$

Now fix some $m \geqslant 1$ and assume for induction that $(*)$ holds for all $0 \leqslant n \leqslant m$. In other words, we assume for all $0 \leqslant n \leqslant m$ that $ax_n + by_n = r_n$. In order to show that $(*)$ holds for $n = m + 1$, we observe from the definitions that

$$\begin{aligned} x_{m+1} &= x_m - q_{m+1}x_{m-1}, \\ y_{m+1} &= y_m - q_{m+1}y_{m-1}, \\ r_{m+1} &= r_m - q_{m+1}r_{m-1}. \end{aligned}$$

Hence we have

$$\begin{aligned} ax_{m+1} + by_{m+1} &= a(x_m - q_m x_{m-1}) + b(y_m - q_m y_{m-1}) \\ &= (ax_m + by_m) - q_{m+1}(ax_{m-1} + by_{m-1}) \\ &= r_m - q_{m+1}r_{m-1} \\ &= r_{m+1}. \end{aligned}$$

□

This explains how Alice can find integers d, k satisfying $de = (p-1)(q-1)k + 1$. Since by assumption we have $\gcd(e, (p-1)(q-1)) = 1$, the Extended Euclidean Algorithm will produce integers $x, y \in \mathbb{Z}$ satisfying

$$ex + (p-1)(q-1)y = 1.$$

Then we just rename them $d = x$ and $k = -y$. Next we will use the Extended Euclidean Algorithm for a surprising theoretical purpose. The following famous result also goes back to Euclid's *Elements*, Book X.

Euclid's Lemma

We say that an integer $p \geqslant 2$ is *prime* when it has no divisors other than 1 and itself. If p is prime then for all integers $a, b \in \mathbb{Z}$ we have

$$p|(ab) \quad \Longrightarrow \quad p|a \text{ or } p|b.$$

Remark: This property is **not** true when p is not prime. For example, with $p = 4$, $a = 2$ and $b = 6$ we have $p|(ab)$, but $p \nmid a$ and $p \nmid b$.

The proof is a classic. I ask my algebra students to memorize it, but you don't have to.

Proof. Instead of proving that $p|(ab)$ implies $p|a$ or $p|b$, we will prove that $p|(ab)$ and $p \nmid a$ imply $p|b$, which is logically equivalent.[42] So let us suppose that $p|(ab)$ with p prime and $p \nmid a$. In this case I claim that $\gcd(a, p) = 1$. Indeed, if $d = \gcd(a, p)$ then we must have $d|p$, which implies that $d = 1$ or $d = p$ because p is prime. On the other hand, we must have $d|a$, which excludes the possibility $d = p$ because we assumed that $p \nmid a$. Hence $d = 1$.

Since $\gcd(a, p) = 1$, the Extended Euclidean Algorithm will produce integers $x, y \in \mathbb{Z}$ satisfying $ax + py = 1$. Finally, since $p|(ab)$ we can write $ab = pk$ for some $k \in \mathbb{Z}$, and hence

$$\begin{aligned} ax + py &= 1 \\ (ax + py)b &= b \\ abx + pby &= b \\ pkx + pby &= b \\ p(kx + by) &= b. \end{aligned}$$

It follows that $p|b$ as desired. □

[42] The principle here is that $P \Rightarrow (Q \vee R)$ is the same as $(P \wedge \neg Q) \Rightarrow R$, which you can verify with a truth table.

Typically, Euclid's Lemma is used[43] to prove the "Fundamental Theorem of Arithmetic", which says that every integer has a unique factorization as a product of primes. We don't have any special need for that fact so I won't prove it here. Instead I will use Euclid's Lemma to prove Fermat's Little Theorem, which is the key to RSA.

For the remainder of this section it is convenient to change notation slightly, from the language of remainders to the language of modular arithmetic.

Definition of Modular Arithmetic

Fix an integer $n \geqslant 1$, which we will call the *modulus*. Then for any integers $a, b \in \mathbb{Z}$ we define the notation

$$a \equiv b \pmod{n} \quad \Longleftrightarrow \quad n|(a-b).$$

In this case we will say that *a is congruent to b modulo n*. This is equivalent to saying that a and b have the same remainder when divided by n:

$$a \equiv b \pmod{n} \quad \Longleftrightarrow \quad (a \,\mathbf{rem}\, n) = (b \,\mathbf{rem}\, n).$$

The notation "$\equiv$" was introduced by Gauss in the *Disquisitiones Arithmeticae* (1801), which launched the modern era of number theory. It is a very convenient notation because it behaves like an equals sign. That is, we have the following properties:

(i) For all $a \in \mathbb{Z}$ we have $a \equiv a \pmod{n}$.

(ii) For all $a, b \in \mathbb{Z}$ we have $a \equiv b \pmod{n}$ if and only if $b \equiv a \pmod{n}$.

(iii) If $a \equiv b \pmod{n}$ and $b \equiv c \pmod{n}$ then we have $a \equiv c \pmod{n}$.

(iv) If $a \equiv a' \pmod{n}$ and $b \equiv b' \pmod{n}$ then we have $a+b \equiv a'+b'$ and $ab \equiv a'b' \pmod{n}$.

Of course, these properties need to be checked. The first three properties are boring but the fourth is a bit tricky. Assume that $a \equiv a'$ and $b \equiv b' \pmod{n}$. By definition this means that $a - a'$ and $b - b'$ are multiplies of n. Let's say $a - a' = nk$ and $b - b' = n\ell$. It follows that

$$ab = (a' + nk)(b' + n\ell) = a'b' + n(a'\ell + kb' + nk\ell),$$

which shows that $ab - a'b'$ is a multiple of n and hence $ab \equiv a'b' \pmod{n}$. We will see that this property simplifies many calculations.

Now we will state Fermat's theorem in the language of modular arithmetic.

[43] Indeed, this is how Euclid used it.

Fermat's Little Theorem

Let $a, p \in \mathbb{Z}$ with $p \geqslant 2$ prime and $p \nmid a$. Then we have

$$a^{p-1} \equiv 1 \pmod{p}.$$

For example, take $a = 2$ and $p = 11$. To see that $2^{10} \equiv 1 \pmod{11}$ we will work our way through the powers of 2. Note that each step is based on the previous step so the numbers never become too large. This is a consequence of property (iv) above.[44] To save space, we report the modulus only at the end of the computations.

$$\begin{aligned}
2^1 &\equiv 2,\\
2^2 &\equiv 4,\\
2^3 &\equiv 8,\\
2^4 &\equiv 16 \equiv 5,\\
2^5 &\equiv 2^4 \cdot 2 \equiv 5 \cdot 2 \equiv 10,\\
2^6 &\equiv 2^5 \cdot 2 \equiv 10 \cdot 2 \equiv 20 \equiv 9,\\
2^7 &\equiv 2^6 \cdot 2 \equiv 9 \cdot 2 \equiv 18 \equiv 7,\\
2^8 &\equiv 2^7 \cdot 2 \equiv 7 \cdot 2 \equiv 14 \equiv 3,\\
2^9 &\equiv 2^8 \cdot 2 \equiv 3 \cdot 2 \equiv 6,\\
2^{10} &\equiv 2^9 \cdot 2 \equiv 6 \cdot 2 \equiv 12 \equiv 1,\\
2^{11} &\equiv 2^{10} \cdot 2 \equiv 1 \cdot 2 \equiv 2 \pmod{11}.
\end{aligned}$$

This is quite mysterious. The powers of 2 reduced mod 11 seem to bounce around with no obvious pattern. Yet somehow we have $2^{10} \equiv 1 \pmod{11}$. After this the sequence of powers will repeat because

$$2^{k+10} \equiv 2^k \cdot 2^{10} \equiv 2^k \cdot 1 \equiv 2^k \pmod{11}.$$

For example, we can easily compute 2^{557} by hand:

$$2^{557} \equiv 2^{10 \cdot 55 + 7} \equiv (2^{10})^{55} \cdot 2^7 \equiv (1)^{55} \cdot 2^7 \equiv 2^7 \equiv 7 \pmod{11}.$$

Actually Fermat did not leave behind a proof of his little theorem, and it was first proved over 100 years later by Euler.[45] Today we can view Fermat's Little Theorem as a simple theorem of "group theory", which is discussed in MTH 461 and MTH 561/562. If you are seriously

[44] This kind of simplification is what allows Alice and Bob to compute m^e and c^d, even when the exponents e and d are quite large.

[45] Scientific journals did not exist in Fermat's time. We mostly know of his work through his correspondence with other scientists.

interested in cryptography then you should definitely study groups. Here I will present Euler's first proof, since it uses induction and binomial coefficients. It does require one lemma, which you will prove on the homework.

Freshman's Dream[46]

Let $p \geqslant 2$ be prime. Then for all integers $0 < k < p$ we have

$$\binom{p}{k} \equiv 0 \pmod{p}.$$

It follows from this that for all integers a, b we have

$$(a+b)^p \equiv a^p + b^p \pmod{p}.$$

Euler's Proof of Fermat's Little Theorem. Let $p \geqslant 2$ be prime. We will prove by induction that $n^p \equiv n \pmod{p}$ for all $n \geqslant 1$. The base case says that $1^p \equiv 1 \pmod{p}$, which is true. Now fix some $n \geqslant 1$ and assume that $n^p \equiv n \pmod{p}$. Then we also have

$$\begin{aligned} (n+1)^p &\equiv n^p + 1^p && \text{Freshman's Dream} \\ &\equiv n^p + 1 && 1^p \equiv 1 \pmod{p} \\ &= n+1 \pmod{p}. && \text{induction} \end{aligned}$$

Thus we have shown that $n^p \equiv n \pmod{p}$ for all integers $n \geqslant 1$. In the case $p \nmid n$ we will show moreover that $n^{p-1} \equiv 1 \pmod{p}$. To do this we note as in the proof of Euclid's Lemma that $p \nmid n$ implies $\gcd(n,p) = 1$, so from the Extended Euclidean Algorithm we can find integers $x, y \in \mathbb{Z}$ satisfying

$$nx + py = 1.$$

Since $p \equiv 0 \pmod{p}$, this equation implies[47]

$$\begin{aligned} nx &\equiv 1 - py \\ &\equiv 1 - 0y \\ &\equiv 1 \pmod{p}. \end{aligned}$$

Finally, we multiply both sides of the congruence $n^p \equiv n \pmod{p}$ by x to obtain

$$\begin{aligned} n^p &\equiv n \\ n^p x &\equiv nx \end{aligned}$$

[46] I really don't like this name, but it's in Wikipedia so I'll go with it.

[47] We say that x is the *inverse* of n mod p. This is not quite the same as division, but it is just as useful.

$$\begin{aligned} n^{p-1}(nx) &\equiv nx \\ n^{p-1}1 &\equiv 1 \\ n^{p-1} &\equiv 1 \pmod{p}. \end{aligned}$$

□

It is worth isolating the last step of that proof. Given the congruence $n^p \equiv n \pmod{p}$ we were able to "divide both sides by n" to obtain $n^{p-1} \equiv 1 \pmod{p}$. However, division is not always possible in modular arithmetic. For example, we have $2 \cdot 3 \equiv 0 \pmod{6}$. Suppose we could divide both sides by 2. Then we would obtain

$$3 \equiv \frac{2 \cdot 3}{2} \equiv \frac{0}{2} \equiv 0 \pmod{6},$$

which is false. Here is the general theorem that tells us when we can divide mod n.

Division Mod n

Consider integers $a, n \geqslant 1$ and suppose that $\gcd(a, n) = 1$. Then it is possible to "divide by a mod n". To see this, we use the Extended Euclidean Algorithm to find $x, y \in \mathbb{Z}$ satisfying

$$ax + ny = 1.$$

Then since $n \equiv 0 \pmod{n}$, reducing both sides by n gives

$$ax \equiv 1 - ny \equiv 1 - 0y \equiv 1 \pmod{n}.$$

This tells us that in some sense $x \equiv 1/a \pmod{n}$, so "division by by a" is the same as "multiplication by x".

For example, since $\gcd(21, 34) = 1$, we can divide both sides of the following congruence by 21 to solve for c:

$$21c \equiv 5 \pmod{34}.$$

In order to do this, we use the Extended Euclidean Algorithm to find $x, y \in \mathbb{Z}$ such that

$21x + 34y = 1$:[48]

x	y	r
1	0	34
0	1	21
1	−1	13
−1	2	8
2	−3	5
−3	5	3
5	−8	2
−8	13	1

We conclude that $34(-8) + 21(13) = 1$ and hence

$$21 \cdot 13 \equiv 1 \pmod{34}.$$

Finally, we multiply both sides of the congruence $21c \equiv 5 \pmod{34}$ by 13 to obtain

$$\begin{aligned} 21c &\equiv 5 \\ 13 \cdot 21c &\equiv 13 \cdot 5 \\ 1c &\equiv 65 \\ c &\equiv -3 \\ c &\equiv 31 \pmod{34}. \end{aligned}$$

As you see, there are many equivalent ways to state the answer. We say that $c \equiv 31 \pmod{34}$ is the *standard form* because 31 is a valid remainder mod 34. In summary: if $21c$ has remainder 5 when divided by 34 then c has remainder 31 when divided by 34.

Now we have the necessary ingredients to prove that the RSA cryptosystem works. Recall that Bob encodes his message by raising m to the power of the encryption exponent, and reducing mod n:

$$c = m^e \,\mathbf{rem}\, n.$$

Then Alice decrypts the message by raising c to the power of the decryption exponent, and reducing mod n:

$$m' = c^d \,\mathbf{rem}\, n.$$

We need to prove that m' is equal to Bob's original message m. The keep the proof clean it is convenient to isolate the following lemma.

Lemma for RSA

[48] Here I have chosen 12 and 34 to be consecutive Fibonacci numbers, so all of the quotients are 1.

Let $p, q \in \mathbb{Z}$ be any integers satisfying $\gcd(p, q) = 1$.[49] Then for any integer $n \in \mathbb{Z}$ we have

$$p|n \text{ and } q|n \quad \Longrightarrow \quad (pq)|n.$$

The proof uses the same trick as in Euclid's Lemma.

Proof of the lemma. Since $\gcd(p, q) = 1$ we can find $x, y \in \mathbb{Z}$ such that $px + qy = 1$. Now assume that $p|n$ and $q|n$; say $pk = n$ and $q\ell = n$. It follows that

$$\begin{aligned} 1 &= px + qy \\ n &= n(px + qy) \\ &= npx + nqy \\ &= (q\ell)px + (pk)qy \\ &= (pq)(\ell x + ky), \end{aligned}$$

and hence $(pq)|n$. □

Proof that RSA works. Recall the setup:

- p and q are distinct primes, so that $\gcd(p, q) = 1$,
- $n = pq$,
- $de = (p - 1)(q - 1)k + 1$,
- $c = m^e \,\mathbf{rem}\, n$.

Given this, our goal is to show that $c^d \,\mathbf{rem}\, n = m$. Translating into the language of modular arithmetic, we will show that[50]

$$c^d \equiv m \pmod{n}.$$

Actually, we will prove an equivalent statement. Since $c \equiv m^e \pmod{n}$, $n = pq$ and $de = (p - 1)(q - 1)k + 1$, we observe that

$$\begin{aligned} c^d \equiv m \pmod{n} &\Leftrightarrow n|(c^d - m) \\ &\Leftrightarrow n|(m^{de} - m) \\ &\Leftrightarrow (pq)|(m(m^{de-1} - 1)) \\ &\Leftrightarrow (pq)|(m(m^{(p-1)(q-1)k} - 1)). \end{aligned}$$

That looks weird but it's convenient for the proof. First we will show that

$$p|(m(m^{(p-1)(q-1)k} - 1)).$$

[49] Jargon: Such pairs of integers are called *coprime*, or *relatively prime*.

[50] Since $0 \leqslant m < n$, this is equivalent to showing that m is the remainder of c^d when divided by n.

Indeed, if $p|m$ then there is nothing to show. Otherwise, if $p \nmid m$ then Fermat's Little Theorem tells us that $m^{p-1} \equiv 1 \pmod{p}$, and hence

$$m^{(p-1)(q-1)k} \equiv (m^{p-1})^{(q-1)k} \equiv 1^{(q-1)k} \equiv 1 \pmod{p}.$$

But this is equivalent to $p|(m^{(p-1)(q-1)k} - 1)$, so again we have $p|(m(m^{(p-1)(q-1)k} - 1))$. A similar proof shows that

$$q|(m(m^{(p-1)(q-1)k} - 1)).$$

Finally, since $\gcd(p, q) = 1$, the Lemma for the RSA implies that

$$(pq)|(m(m^{(p-1)(q-1)k} - 1)).$$

□

As you see, this was not an easy proof. For the purpose of this class I will only ask you to prove much smaller results, such as the Freshman's Dream above. I just figured that a computer scientist should see the details of RSA once in their life.

3.7 Worked Exercises

3.1. Here is a false proof. Find the mistake.

Claim. The following statement is true for all $n \geqslant 0$:

$$P(n) = \text{“if } a, b \in \mathbb{Z} \text{ satisfy } a, b \geqslant 0 \text{ and } n = \max(a, b) \text{ then } a = b\text{”}.$$

Proof. Clearly $P(0)$ is true because $a, b \geqslant 0$ and $\max(a, b) = 0$ imply $a = b = 0$. Now fix some $n \geqslant 0$ and assume for induction that $P(n)$ is true. In order to prove that $P(n + 1)$ is also true we consider any integers with $a, b \geqslant 0$ and $\max(a, b) = n + 1$. But then we have $\max(a - 1, b - 1) = n$ and $P(n)$ implies that $a - 1 = b - 1$, hence $a = b$. □

To find the mistake we will closely examine the proof that $P(0)$ implies $P(1)$. So consider any $a, b \geqslant 0$ with $\max(a, b) = 1$. Then certainly it is true that $\max(a - 1, b - 1) = 0$. But it does not necessarily follow from $P(0)$ that $a - 1 = b - 1$. Indeed, consider the case when $a = 0$ and $b = 1$. Then $\max(a, b) = 1$ and $\max(a - 1, b - 1) = 0$, but $a \neq b$. The problem here is that $a - 1 < 0$ and hence $P(0)$ **does not apply** to the numbers $a - 1$ and $b - 1$. In general, our argument that $P(n)$ implies $P(n + 1)$ is wrong because it might be the case that $a - 1 = -1$ or $b - 1 = -1$.

3.2. Given $a, b \in \mathbb{Z}$ we define the following notation:

$$\text{“}a|b\text{”} = \text{“}a \text{ divides } b\text{”} = \text{“}\exists k \in \mathbb{Z}, ak = b\text{”}.$$

We say that $n \in \mathbb{N}$ is *not prime* if there exist $a, b \in \mathbb{Z}$ with $n = ab$ and $a, b \in \{2, 3, \ldots, n - 1\}$. (We say that a and b are *proper factors* of n.) Now consider the following statement:

Every natural number $n \geqslant 2$ is divisible by some prime number.

(a) Prove the statement by strong induction.

(b) Prove the statement by well-ordering.

(a) **Proof by Strong Induction.** The base case is $b = 2$. Note that $P(2)$ is true because 2 is prime and $2|2$. Now assume for induction that the statements $P(2), P(3), \ldots, P(n)$ are all true. In this case we want to show that $n + 1$ is has some prime factor. If $n + 1$ is itself prime then we are done because $(n + 1)|(n + 1)$. So let us assume that $n + 1$ is not prime. By definition this means that we can write

$$n + 1 = ab \quad \text{where } a, b \in \{2, \ldots, n - 1\}.$$

But then by our induction hypothesis each of a and b has a prime factor. In particular we have $p|a$ for some prime p. Then we can write $a = pk$ for some $k \in \mathbb{Z}$ and

$$n + 1 = ab = (pk)b = p(kb).$$

It follows that $p|(n + 1)$, and hence $n + 1$ has a prime factor. □

(b) **Proof by Well-Ordering.** Consider the set of "criminals":

$$S := \{n \geqslant 2 : n \text{ is not divisible by any prime}\}.$$

Our goal is to show that S is empty, so assume for contradiction that S is not empty. Then since S is bounded below by 2 we know by well-ordering that there exists a least element $m \in S$, called a "minimal criminal". Since $m|m$ and since m is not divisible by any prime, we know that m is not prime. Therefore we must have

$$m = ab \quad \text{where } a, b \in \{2, \ldots, m - 1\}.$$

But then since $a < m$ and $b < m$ we know that a and b are **not criminals**. In other words each of a and b is divisible by some prime. In particular, we have $p|a$ for some prime p. Then we conclude as above that $p|m$. But this contradicts the fact that m has no prime factor. □

3.3. Convert the decimal number 123456789 into the following base systems:

(a) Binary $\{0, 1\}$

(b) Ternary $\{0, 1, 2\}$

(c) Hexadecimal $\{0, 1, \ldots, 9, A, B, \ldots, F\}$

I programmed the algorithm into my computer. Here are the results:

$$123456789 = (111010110111100110100010101)_2$$

$$
\begin{aligned}
&= (22121022020212200)_3 \\
&= (13112330310111)_4 \\
&= (223101104124)_5 \\
&= (20130035113)_6 \\
&= (3026236221)_7 \\
&= (726746425)_8 \\
&= (277266780)_9 \\
&= (123456789)_{10} \\
&= (63762A05)_{11} \\
&= (35418A99)_{12} \\
&= (1C767471)_{13} \\
&= (12579781)_{14} \\
&= (AC89BC9)_{15} \\
&= (75BCD15)_{16}
\end{aligned}
$$

3.4. Convert the decimal numbers 12 and 23 into binary. Multiply them in binary. Then convert the result back into decimal notation.

We have $12 = (1100)_2$ and $23 = (10111)_2$. Here is the multplication algorithm:

$$
\begin{array}{cccccccccc}
 & & & & & 1 & 0 & 1 & 1 & 1 \\
\times & & & & & & 1 & 1 & 0 & 0 \\
\hline
 & & & & & 0 & 0 & 0 & 0 & 0 \\
 & & & & 0 & 0 & 0 & 0 & 0 & \\
 & & & 1 & 0 & 1 & 1 & 1 & & \\
+ & 1 & 0 & 1 & 1 & 1 & & & & \\
\hline
1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 &
\end{array}
$$

The hardest part is carrying. Then we convert the result back into decimal notation:

$$
\begin{aligned}
(100010100)_2 &= 1 \cdot 2^8 + 1 \cdot 2^4 + 1 \cdot 2^2 \\
&= 256 + 16 + 2 \\
&= 276.
\end{aligned}
$$

3.5. Euclidean Algorithm.

(a) Apply the Euclidean Algorithm to compute the gcd of 3094 and 2513.

(b) Repeat the same sequence of steps to find the continued fraction expansion of 3094/2513:

$$\frac{3094}{2513} = q_1 + \frac{1}{q_2 + \frac{1}{q_3 + \frac{1}{q_4 + \cdots}}}.$$

(a) We begin by dividing 3094 by 2513 to get remainder 581. Then we replace the pair $(3094, 2513)$ by the pair $(2513, 581)$ and repeat:

$$\begin{aligned}
\mathbf{3094} &= 1 \cdot \mathbf{2513} + \mathbf{581} \\
\mathbf{2513} &= 4 \cdot \mathbf{581} + \mathbf{189} \\
\mathbf{581} &= 3 \cdot \mathbf{189} + \mathbf{14} \\
\mathbf{189} &= 13 \cdot \mathbf{14} + \mathbf{7} \\
\mathbf{14} &= 2 \cdot \mathbf{7} + \mathbf{0}
\end{aligned}$$

The last nonzero remainder is $7 = \gcd(3094, 2513)$.

(b) The sequence of quotients in part (a) tells us the continued fraction expansion:

$$\frac{3094}{2513} = 1 + \frac{1}{4 + \frac{1}{3 + \frac{1}{13 + \frac{1}{2}}}}.$$

3.6. $\sqrt{2}$ is Irrational. If a and b are integers then the Euclidean Algorithm guarantees that the continued fraction expansion of a/b is **finite**. Prove that

$$\sqrt{2} = 1 + \frac{1}{1 + \sqrt{2}}$$

and use this to show that the continued fraction expansion of $\sqrt{2}$ is **infinite**. It follows that $\sqrt{2}$ is not a fraction of integers.

Note that

$$\begin{aligned}
(\sqrt{2} - 1)(\sqrt{2} + 1) &= \sqrt{2}^2 - 1^2 \\
(\sqrt{2} - 1)(\sqrt{2} + 1) &= 2 - 1 \\
(\sqrt{2} - 1)(\sqrt{2} + 1) &= 1 \\
\sqrt{2} - 1 &= \frac{1}{\sqrt{2} + 1} \\
\sqrt{2} &= 1 + \frac{1}{1 + \sqrt{2}}.
\end{aligned}$$

Therefore we have

$$\begin{aligned}
\sqrt{2} &= 1 + \frac{1}{1+\sqrt{2}} \\
&= 1 + \frac{1}{1+1+\frac{1}{1+\sqrt{2}}} \\
&= 1 + \frac{1}{2+\frac{1}{1+\sqrt{2}}} \\
&= 1 + \frac{1}{2+\frac{1}{1+1+\frac{1}{1+\sqrt{2}}}} \\
&= 1 + \frac{1}{2+\frac{1}{2+\frac{1}{1+\sqrt{2}}}} \\
&= 1 + \frac{1}{2+\frac{1}{2+\frac{1}{2+\cdots}}}.
\end{aligned}$$

Since the process never stops we conclude that $\sqrt{2}$ is an irrational number.

4 Principles of Counting

In the first chapter we defined the numbers $\binom{n}{k}$ for integers $n \geqslant k \geqslant 0$ by the following boundary conditions and recurrence relation:

$$\binom{n}{k} := \begin{cases} 1 & \text{if } k = 0 \text{ or } k = n, \\ \binom{n-1}{k-1} + \binom{n-1}{k} & \text{otherwise.} \end{cases}$$

Recall that these numbers can be computed by hand via *Pascal's triangle*:

$$\begin{array}{ccccccccccc}
 & & & & \binom{0}{0} & & & & & & \\
 & & & \binom{1}{0} & & \binom{1}{1} & & & & & \\
 & & \binom{2}{0} & & \binom{2}{1} & & \binom{2}{2} & & & & \\
 & \binom{3}{0} & & \binom{3}{1} & & \binom{3}{2} & & \binom{3}{3} & & & \\
\binom{4}{0} & & \binom{4}{1} & & \binom{4}{2} & & \binom{4}{3} & & \binom{4}{4} & &
\end{array}
\quad = \quad
\begin{array}{ccccccccccc}
 & & & & 1 & & & & & & \\
 & & & 1 & & 1 & & & & & \\
 & & 1 & & 2 & & 1 & & & & \\
 & 1 & & 3 & & 3 & & 1 & & & \\
1 & & 4 & & 6 & & 4 & & 1 &
\end{array}$$

Then we used the recursive definition to prove the following two theorems by induction:

- For any number x and for any integer $n \geqslant 0$ we have

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

- For any integers $n \geqslant k \geqslant 0$ we have

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

where the *factorial* notation is defined recursively by

$$n! := \begin{cases} 1 & \text{if } n = 0, \\ n \cdot (n-1)! & \text{if } n \geqslant 0. \end{cases}$$

Taken together, these two theorems are often called the *binomial theorem*:

$$(1+x)^n = \sum_{k=0}^{n} \frac{n!}{k!(n-k)!}\, x^k.$$

For this reason, the numbers $\binom{n}{k}$ are also called *binomial coefficients*.

So far all of these ideas can be viewed as pure algebra. However, when I read the symbol $\binom{n}{k}$ out loud I say "n choose k". In this chapter I will explain what the word "choose" means, and I will show you the many ways that binomial coefficients show up in counting problems.

4.1 Counting Ordered Selections

In this section and the next we will count the number of ways to choose k objects from a set of n objects. We will get different answers, depending on whether the choices are ordered or unordered, and whether we are allowed to select a given object more than once. However, all of our counting methods will depend on the following fundamental principle.

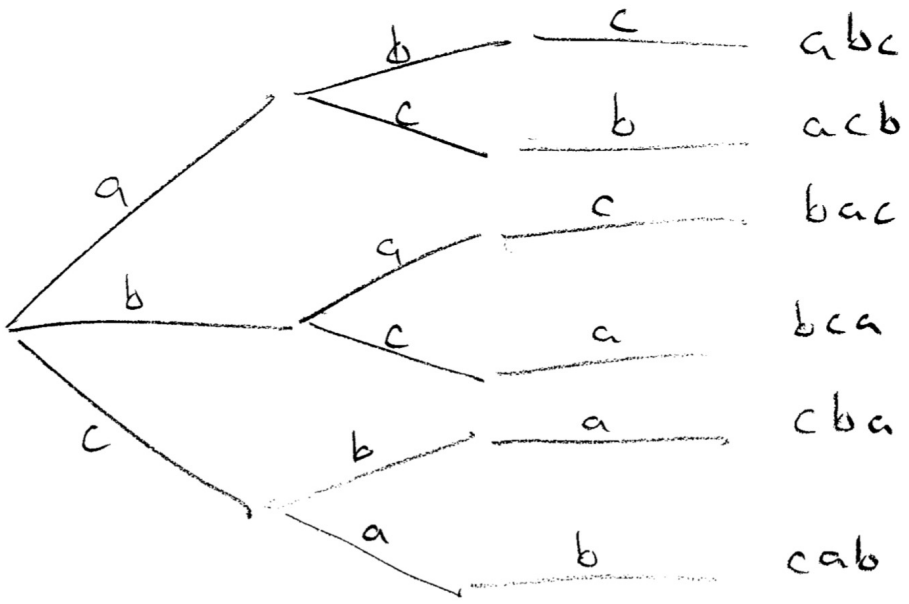
The Multiplication Principle

When a sequence of choices is made, the number of possibilities multiply.

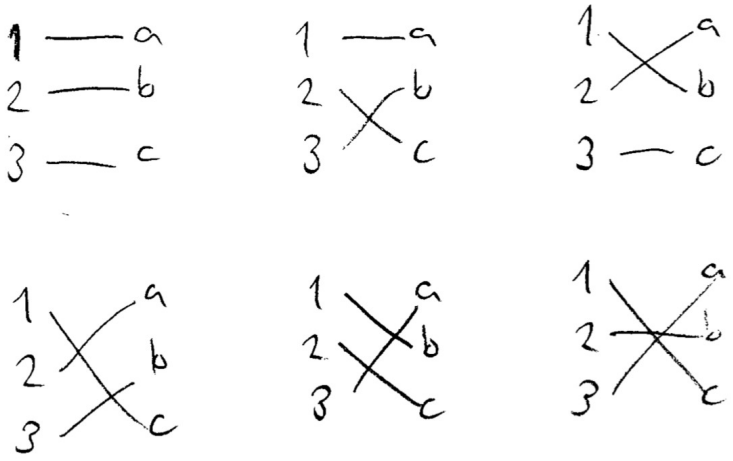
To illustrate what this means, let us count the number of ways to put the symbols a, b, c in order. We can think of this problem as choosing 3 things from the set $\{a, b, c\}$ where **order matters and repetition is not allowed**. Since this is a sequence of choices we can use the multiplication principle. There are 3 ways to choose the first thing, then there are 2 ways to choose the second thing (because one thing is not allowed). Finally, there is only 1 way to choose the third thing (because two things are not allowed):

$$\underbrace{3}_{\text{1st choice}} \times \underbrace{2}_{\text{2nd choice}} \times \underbrace{1}_{\text{3rd choice}} = 3 \cdot 2 \cdot 1 = 6$$

These choices are called *permutations*. We can view each permutation as a path in a branching tree:



The tree structure is the reason that the possibilities multiply. On the other hand, we can think of each permutation as an injective function from the set $\{1, 2, 3\}$ to $\{a, b, c\}$ where 1 gets sent to the first choice, 2 gets sent to the second choice and 3 gets sent to the third choice:



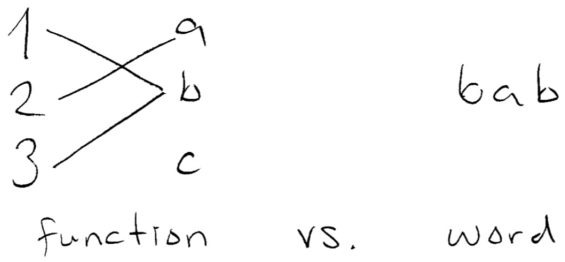
Sometimes the most difficult part of a counting problem is to recognize when two situations are really the same.

Now let's modify the problem to allow repeated letters. In other words, we want to choose

3 things from the set $\{a, b, c\}$, where **order maters and repetition is allowed**. This time the number of choices is

$$\underbrace{3}_{\text{1st choice}} \times \underbrace{3}_{\text{2nd choice}} \times \underbrace{3}_{\text{3rd choice}} = 3^3 = 27.$$

We can view the choices as "words of length 3 from the alphabet $\{a, b, c\}$" or we can view them as arbitrary functions from $\{1, 2, 3\}$ to $\{a, b, c\}$ (not necessarily injective). I don't want to draw all 27 choices, but here is a corresponding pair:



Similarly, the number of ways to choose 4 things from the set $\{a, b, c\}$ when order matters and repetition is allowed is

$$\underbrace{3}_{\text{1st choice}} \times \underbrace{3}_{\text{2nd choice}} \times \underbrace{3}_{\text{3rd choice}} \times \underbrace{3}_{\text{4th choice}} = 3^4 = 81.$$

This is the number of arbitrary functions from $\{1, 2, 3, 4\}$ to $\{a, b, c\}$, or equivalently the number of words of length 4 from the alphabet $\{a, b, c\}$.

How about the number of ways to choose 4 things from the set $\{a, b, c\}$ when **order matters and repetition is not allowed**? This time the number of choices is

$$\underbrace{3}_{\text{1st choice}} \times \underbrace{2}_{\text{2nd choice}} \times \underbrace{1}_{\text{3rd choice}} \times \underbrace{0}_{\text{4th choice}} = 3 \cdot 2 \cdot 1 \cdot 0 = 0.$$

In other words, it is impossible to choose 3 things from a set of 4 when repetition is not allowed. Based on these examples we have the following two theorems.

Counting Words

Let $n, k \geqslant 0$ and consider the following collections:

- Selections of k from n things when order matters and repetition is allowed.
- Words of length k from an alphabet of size n.
- Functions from a set of size k to a set of size n.

I claim that each of these collections is counted by the number n^k.

Proof. Each object in each collection is determined by a sequence of k choices in which there are n possibilities for each choice. Thus the total number of possibilities is

$$\underbrace{n}_{\text{1st choice}} \times \underbrace{n}_{\text{2nd choice}} \times \cdots \times \underbrace{n}_{k\text{th choice}} = n^k.$$

Counting Permutations

Let $n, k \geqslant 0$ and consider the following collections:

- Selections of k from n things when order matters and repetition is **not** allowed.
- Words with no repeated letters of length k from an alphabet of size n
- Injective functions from a set of size k to a set of size n.

I claim that each of these collections is counted by the number

$$_nP_k := n(n-1)(n-2)\cdots(n-k+2)(n-k+1).$$

Proof. Each object in each collection is determined by a sequence of k choices, with $n = n-0$ possibilities for the 1st choice, then $n-1$ possibilities for the 2nd choice, ..., then $n-(i-1)$ possibilities for the ith choice,...,then $n-(k-1) = n-k+1$ possibilities for the kth choice. Thus the total number of possibilities is

$$\underbrace{n}_{\text{1st choice}} \times \underbrace{n-1}_{\text{2nd choice}} \times \cdots \times \underbrace{n-k+1}_{k\text{th choice}} = n(n-1)(n-2)\cdots(n-k+2)(n-k+1).$$

□

Let me make some observations about the numbers $_nP_k$:

- If $k > n$ then one of the factors in the product is zero, hence $_nP_k = 0$. In this case it is **impossible** to choose k things from n things without repetition.
- If $k = n$ then we obtain $_nP_n = n(n-1)(n-2)\cdots(n-n+1) = n!$. These selections are called *permutations*, and this is the reason for the P in the number $_nP_k$.

- If $0 \leqslant k < n$ then the formula for ${}_nP_k$ can be simplified by multiplying by the fraction $(n-k)!/(n-k)!$ to get

$$\begin{aligned} {}_nP_k &= n(n-1)(n-2)\cdots(n-k+2)(n-k+1) \cdot \frac{(n-k)(n-k-1)\cdots 3\cdot 2\cdot 1}{(n-k)(n-k-1)\cdots 3\cdot 2\cdot 1} \\ &= \frac{n(n-1)(n-2)\cdots 3\cdot 2\cdot 1}{(n-k)(n-k-1)\cdots 3\cdot 2\cdot 1} \\ &= \frac{n!}{(n-k)!}. \end{aligned}$$

If you want, we could also define ${}_nP_k = 0$ when $k < 0$. Thus for all $n, k \in \mathbb{Z}$ with $n \geqslant 0$ we have the following definition of the "permutation numbers":

$${}_nP_k := \begin{cases} 0 & \text{if } k < 0 \text{ or } k > n, \\ n!/(n-k)! & \text{if } 0 \leqslant k \leqslant n. \end{cases}$$

4.2 Counting Unordered Selections

In general it is much harder to count unordered selections of things. For this we need tricks.

Let us begin by counting unordered selections of k things from n things when repetition is not allowed. For some reason these are called *combinations*, but I prefer to call them *subsets*. Our goal is to find a closed formula for the following "combination numbers":

$$\begin{aligned} {}_nC_k &:= \#\{\text{ways to choose } k \text{ unordered things from } n \text{ things without repetition}\} \\ &= \#\{\text{subsets of size } k \text{ from a set of size } n\} \\ &= \#\{\text{binary strings of length } n \text{ with } k \text{ copies of 1}\}. \end{aligned}$$

In order to develop intuition, let's run an experiment. Here are all of the subsets of the set $\{1,2,3,4\}$ and binary strings of length 4, arranged by size and the number of 1s:

binary strings with k copies of 1	subsets of size k	${}_4C_k$
1111	$\{1,2,3,4\}$	1
1110, 1101, 1011, 0111	$\{1,2,3\}, \{1,2,4\}, \{1,3,4\}, \{2,3,4\}$	4
1100, 1010, 1001, 0110, 0101, 0011	$\{1,2\}, \{1,3\}, \{1,4\}, \{2,3\}, \{2,4\}, \{3,4\}$	6
1000, 0100, 0010, 0001	$\{1\}, \{2\}, \{3\}, \{4\}$	4
0000	$\varnothing$	1

Based on this example we might make the following conjecture:

$$\boxed{{}_nC_k = \binom{n}{k} ?}$$

It turns out this conjecture is true.

Counting Subsets (Combinations)

Let $n \geqslant k \geqslant 0$ and consider the following collections:

- Unordered selections of k things n things when repetition is not allowed.
- Subsets of size k from a set of size n.
- Binary strings with k copies of 1 and $n-k$ copies of 0.

Let ${}_nC_k$ be the number of elements in each collection. Then I claim that

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

I will give two proofs. First a bad proof, then a good proof.

Bad Proof (Induction). Consider the set $\{1, 2, \ldots, n\}$ and let ${}_nC_k$ be the number of subsets of size k. Since the binomial coefficients are defined by recursion we will verify that the combination numbers ${}_nC_k$ satisfy the same recursion.

The boundary conditions are true because for any $n \geqslant 0$ we have ${}_nC_0 = 1 = \binom{n}{0}$ and ${}_nC_n = 1 = \binom{n}{n}$. Indeed, there is only one way to choose none of the elements, and there is only one way to choose all of the elements.

Now suppose that $0 < k < n$. In this case we will prove that ${}_nC_k = {}_{n-1}C_{k-1} + {}_{n-1}C_k$. To see this, let S be the set of subsets of size k from the set $\{1, 2, \ldots, n\}$, so that $\#S = {}_nC_k$. Now we will divide the collection S into two subcollections, depending on whether the symbol n is in the subset:

$$ \begin{aligned} S' &:= \{A \subseteq \{1, \ldots, n\} : \#A = k \text{ and } n \in A\}, \\ S'' &:= \{A \subseteq \{1, \ldots, n\} : \#A = k \text{ and } n \notin A\}. \end{aligned} $$

For example, if $n = 5$ and $k = 3$ then we have

$$ \begin{aligned} S' &= \{\{1,2,5\}, \{1,3,5\}, \{1,4,5\}, \{2,3,5\}, \{2,4,5\}, \{3,4,5\}\}, \\ S'' &= \{\{1,2,3\}, \{1,2,4\}, \{1,3,4\}, \{2,3,4\}\}. \end{aligned} $$

Notice that $\#S' = {}_2C_4 = 6$ because 5 is already in the subset. In order to fill in the remaining $k - 1 = 2$ elements we must choose a subset of size 2 from $\{1, 2, 3, 4\}$ and there are ${}_2C_4$ ways to do this. Note also that $S'' = {}_3C_4$. Since these subsets don't contain 5 they are just subsets of size $k = 3$ from the remaining elements $\{1, 2, 3, 4\}$.

The same reasoning in general shows that

$$ \#S' = \#\{\text{subsets of size } k \text{ from } \{1, 2, \ldots, n\} \text{ that contain } n\} $$

$$= \#\{\text{subsets of size } k-1 \text{ from } \{1, 2 \ldots, n-1\}\}$$
$$= {}_{n-1}C_{k-1}$$

and

$$\begin{aligned}\#S' &= \#\{\text{subsets of size } k \text{ from } \{1, 2, \ldots, n\} \text{ that don't contain } n\} \\ &= \#\{\text{subsets of size } k \text{ from } \{1, 2 \ldots, n-1\}\} \\ &= {}_{n-1}C_k\end{aligned}$$

Finally, since S is a disjoint union[51] of S' and S'' we conclude that

$${}_nC_k = \#S = \#S' + \#S'' = {}_{n-1}C_{k-1} + {}_{n-1}C_k.$$

□

And here is the good proof. It uses a method called *double counting*. This means that we count the elements of a certain collection in two different ways to obtain a useful equation that we can solve.

Good Proof (Double Counting). Let $n \geqslant k \geqslant 0$ and consider the following set:

$$S := \{\text{ordered sections of } k \text{ things from } n \text{ things when repetition is not allowed}\}.$$

We know from the previous section that $\#S = {}_nP_k = n!/(n-k)!$. On the other hand, we can choose an ordered collection of k things in two steps:

- First choose an unordered collection of k things from n. There are ${}_nC_k$ ways to do this.
- Then put the k things in order. There are $k!$ ways to do this.

From the multiplication principle we obtain

$$\#S = \underbrace{{}_nC_k}_{\text{choose } k \text{ unordered things}} \times \underbrace{k!}_{\text{then put them in order}}.$$

By equating the two expressions for $\#S$ we obtain

$${}_nC_k \cdot k! = {}_nP_k$$
$${}_nC_k = {}_nP_k/k! = \frac{n!/(n-k)!}{k!} = \frac{n!}{k!(n-k)!} = \binom{n}{k}.$$

□

Of course the induction proof also teaches us something, but I find the counting proof more enlightening. In combinatorics[52] we usually prefer a counting proof because it gives us a better understanding of **why** the result is true.

[51] This means that $S' \cup S'' = S$ and $S' \cap S'' = \varnothing$.

[52] The study of counting.

The following table summarizes what we have done in this and the previous section. When selecting k things from a set of n things there are two different parameters, leading to four different answers:

	ordered	unordered
no repetition	${}_nP_k = \frac{n!}{(n-k)!}$	${}_nC_k = \frac{n!}{k!(n-k)!} = \binom{n}{k}$
repetition OK	n^k	?

It only remains for us to count unordered selections with repetition. For example, suppose that you want to buy 5 gallons of ice cream and the possible flavors are $\{c, v, s\}$.[53] In how many ways can you do this?

First we need to come up with some notation to record the choices. Suppose that *ccccc* represents the choice "5 chocolate". Maybe *ccccv* represents the choice "5 chocolate and 1 vanilla". Since order doesn't matter, there are lots of ways to record the same choice:

$$ccccv = cccvc = ccvcc = cvccc = vcccc.$$

So let's agree to put all the c's on the left, v's in the middle and s's on the right. After some trial and error you will find the following 21 choices:

vvvvv	*cvvvv*	*ccvvv*	*cccvv*	*ccccv*	*ccccc*
vvvvs	*cvvvs*	*ccvvs*	*cccvs*	*ccccs*	
vvvss	*cvvss*	*ccvss*	*cccss*		
vvsss	*cvsss*	*ccsss*			
vssss	*csssss*				
sssss					

What is the pattern? With only three flavors we can arrange all the choices in the shape of a triangle. However, with more than three flavors this method won't be so helpful. In general it is much better to use the following trick. We will encode each choice as a binary string where 1's represent gallons of ice cream and 0's represent divisions between the flavors. Here are a few examples:

$$ccvss \leftrightarrow 1101011$$
$$cccvv \leftrightarrow 1110110$$
$$ccccc \leftrightarrow 1111100$$
$$vvvvv \leftrightarrow 0111110$$
$$sssss \leftrightarrow 0011111$$

[53] standing for chocolate, vanilla, strawberry.

And here is the general pattern:

$$\underbrace{11\cdots1}_{\#\text{ of } c\text{'s}}0\underbrace{11\cdots1}_{\#\text{ of } v\text{'s}}0\underbrace{11\cdots1}_{\#\text{ of } s\text{'s}}.$$

Note that every choice corresponds to a unique binary string containing 5 copies of 1 (for the gallons of ice cream) and 2 copies of 0 (for the dividers between the three flavors). On the other hand, we can create such a string by choosing 5 positions for the 1's from 7 possible positions. (Equivalently, we can choose 2 positions for the 0's.) Thus the number of choices is

$$\binom{7}{5} = \binom{7}{2} = \frac{7\cdot 6}{2\cdot 1} = 21.$$

Note that this problem can be encoded in another way. Let $x_c, x_v, x_s \in \mathbb{N}$ be the number of gallons of chocolate, vanilla and strawberry that we purchase. Then we are looking for the number of solutions to the following equation:

$$x_c + x_v + x_s = 5.$$

Here is the general result.

Counting Subsets With Repeated Elements

Let $n, k \geqslant 0$ and consider the following collections:

- Unordered selections of k from n things when repetition is allowed.
- Ways to distribute k identical objects into n labeled boxes.
- Non-negative integer solutions $x_1, x_2, \ldots, x_n \in \mathbb{N}$ to the equation

$$x_1 + x_2 + \cdots + x_n = k.$$

I claim that each of these collections is counted by the number

$$\left(\!\binom{n}{k}\!\right) := \binom{n+k-1}{k}.$$

Proof. In each case we can encode a choice as a binary string containing k copies of 1 and $n-1$ copies of 0. The 1's represent the things, or the objects, or the values of the variables. The 0's represent the divisions between the n different flavors or boxes, or the + symbols. We can create such a binary string by choosing k positions for the 1's among $k + (n-1)$ total positions. Therefore the number of choices is

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

□

The trick of encoding the choices as binary strings is very clever. I would not expect you to come up with this trick by yourself.

We can finally complete our table for the number of ways to choose k things from n things. Note that we started on the bottom left and proceeded clockwise until the bottom right, with the solution of each problem depending on the previous:

	ordered	unordered
no repetition	${}_nP_k = \frac{n!}{(n-k)!}$	${}_nC_k = \frac{n!}{k!(n-k)!} = \binom{n}{k}$
repetition OK	n^k	$\left(\!\binom{n}{k}\!\right) = \binom{n+k-1}{k} = \frac{(n+k-1)!}{k!(n-1)!}$

4.3 Proof by Counting

Sometimes there is more than one way to solve a counting problem. For example, consider the symmetry of the binomial coefficients:

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

We could prove this using pure algebra.

Algebraic Proof.

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

□

But the following counting argument is more meaningful because it gives use a better feeling for **why** the algebraic formula is true.

Counting Proof. Let A be the set of subsets of $\{1, 2, \ldots, n\}$ of size k:

$$A := \{S \subseteq \{1, 2, \ldots, n\} : \#S = k\}.$$

We know from the previous section that $\#A = \binom{n}{k}$. Similarly, we know that $\#B = \binom{n}{n-k}$, where B is the set of subsets of size $n - k$:

$$B := \{S \subseteq \{1, 2, \ldots, n\} : \#S = n - k\}.$$

Now observe that complementation is a bijection between A and B:

$$A \xleftrightarrow{\text{complement}} B.$$

Therefore we have

$$\binom{n}{k} = \#A = \#B = \binom{n}{n-k}.$$

Equivalently, we could think of A as the set of binary strings with k copies of 1 and $n-k$ copies of 0, while B is the set of binary strings with $n-k$ copies of 1 and k copies of 0. Then the bijection is given by switching 0s and 1s:

$$A \xleftrightarrow{\text{switch 0s and 1s}} B.$$

□

Next let's consider the identity

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

We can give an algebraic proof by specializing the binomial theorem.

Algebraic Proof. The binomial theorem tells us that

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

Substitute $x = 1$ to get the result. □

Or we can give a counting proof by interpreting each side as the number of subsets of $\{1, 2, \ldots, n\}$.

Counting Proof. We have a bijection between subsets and binary strings:

$$\{\text{subsets of } \{1, \ldots, n\}\} \longleftrightarrow \{\text{binary strings of length } n\}.$$

Since binary strings are just words from the alphabet $\{0, 1\}$ we conclude that

$$\#\{\text{subsets of } \{1, \ldots, n\}\} = \#\{\text{binary strings of length } n\} = 2^n.$$

On the other hand, we have

$$\#\{\text{subsets of } \{1, \ldots, n\}\} = \sum_{k=0}^{n} \#\{\text{subsets of } \{1, \ldots, n\} \text{ with } k \text{ elements}\} = \sum_{k=0}^{n} \binom{n}{k}.$$

Comparing the two expressions gives the result. □

Here's another one:

$$\binom{n}{0} + \binom{n}{2} + \binom{n}{4} + \cdots = \binom{n}{1} + \binom{n}{3} + \binom{n}{4} + \cdots.$$

Algebraic Proof. Substitute $x = -1$ into the binomial theorem:

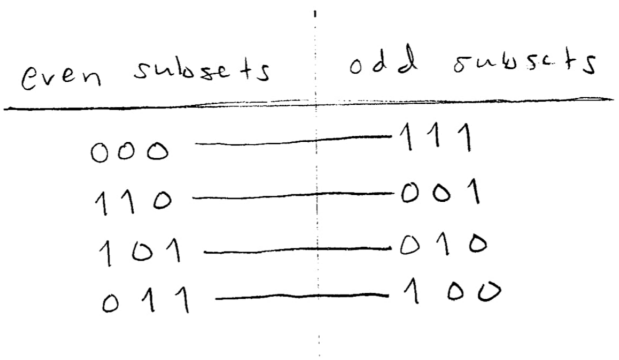
$$\begin{aligned}(1-1)^n &= \binom{n}{0} + \binom{n}{1}(-1) + \binom{n}{2}(-1)^2 + \cdots + \binom{n}{n}(-1)^n \\ 0 &= \binom{n}{0} - \binom{n}{1} + \binom{n}{2} - \cdots \pm \binom{n}{n} \\ \binom{n}{1} + \binom{n}{3} + \cdots &= \binom{n}{0} + \binom{n}{2} + \cdots .\end{aligned}$$

□

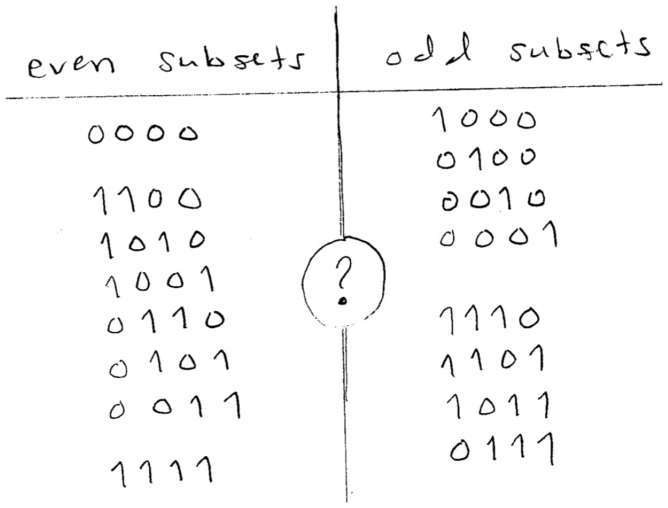
Counting Proof. Let A be the set of subsets of $\{1, \ldots, n\}$ with an even number of elements and let B be the set of subsets of $\{1, \ldots, n\}$ with an odd number of elements. Thus we have

$$\begin{aligned}\#A &= \binom{n}{0} + \binom{n}{2} + \cdots , \\ \#B &= \binom{n}{1} + \binom{n}{3} + \cdots .\end{aligned}$$

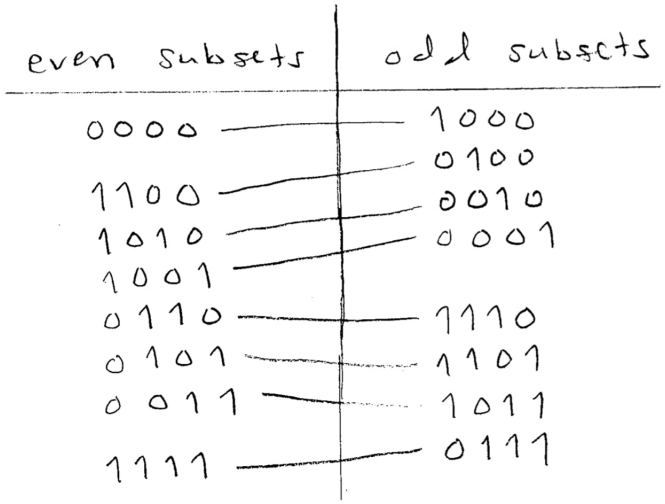
Our goal is to find a bijection between A and B. If n is odd then the complementation map will work. For example, here is the case $n = 3$:



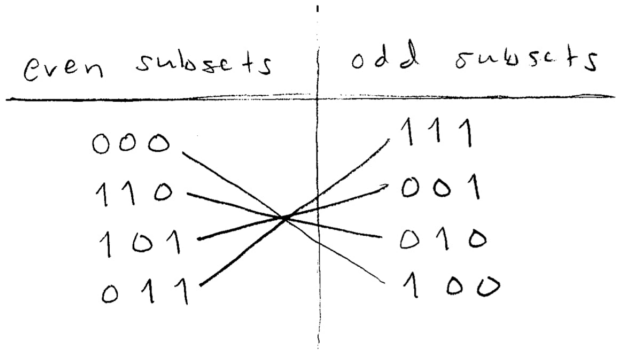
But if n is even then the problem is not so easy. For example, here are the even and odd subsets of $\{1, 2, 3, 4\}$:



We see that there are $1 + 6 + 1 = 8$ even subsets and $4 + 4 = 8$ odd subsets. Can you see a bijection between the two sets? This time complementation does not work, because the complement of an even subset is even and the complement of an odd subset is odd. After a bit of thought we see that "flipping the first bit" gives a bijection:



In retrospect, we see that "flipping the first bit" also works for $n = 3$:

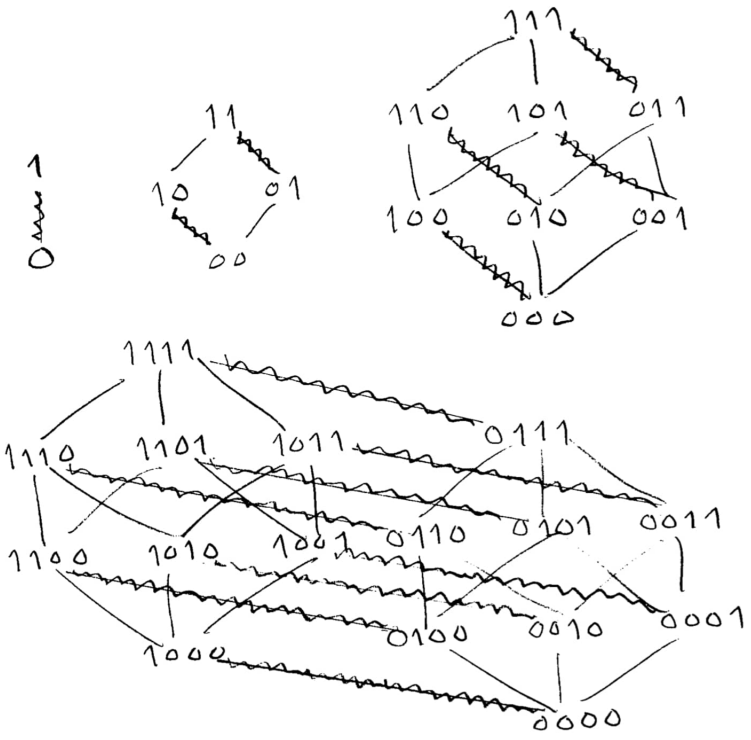


In fact, it doesn't matter which bit we flip. One can check that for any specific $k \in \{1, 2, \ldots, n\}$ we obtain a bijection

$$A \xleftrightarrow{\text{flip the } k\text{th bit}} B,$$

which implies that $\#A = \#B$. □

There is a nice picture of this last proof. I claim that the set of binary strings of length n can be seen as the vertices of "hypercube", where each edge of the hypercube corresponds to flipping one bit. Consider the following picture, where the squiggly lines correspond to "flipping the first bit" and the straight lines correspond to flipping some other bit:



We have seen one type of counting proof called a "bijective proof". Let me end this section by giving an example of another proof technique called "double counting". We will prove the following identity:

$$\binom{i}{j}\binom{j}{k} = \binom{i}{k}\binom{i-k}{j-k} \quad \text{for all } 0 \leqslant i \leqslant j \leqslant k.$$

Algebraic Proof. The left side simplifies to

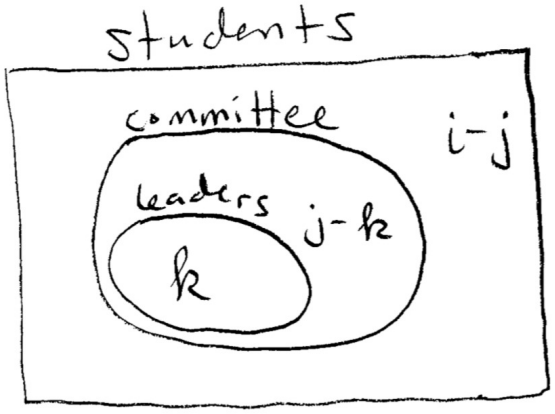
$$\binom{i}{j}\binom{j}{k} = \frac{i!}{\cancel{j}!(i-j)!}\frac{\cancel{j}!}{k!(j-k)!} = \frac{i!}{(i-j)!(j-k)!k!}$$

and the right side simplifies to

$$\binom{i}{k}\binom{i-k}{j-k} = \frac{i!}{k!\cancel{(i-k)!}}\frac{\cancel{(i-k)!}}{(j-k)!((i-\cancel{k})-(j-\cancel{k}))!} = \frac{i!}{(i-j)!(j-k)!k!}.$$

Note that these are the same. □

Proof by Double Counting. Suppose we have a classroom of i students. From these we will choose a committee of j students and then we will choose k members of the committee to be leaders. Here is a picture:



One the one hand, we can first choose j committee members from the i students and then choose k leaders from the j committee members:

$$\#(\text{choices}) = \underbrace{\binom{i}{j}}_{\text{choose committee}} \times \underbrace{\binom{j}{k}}_{\text{then choose leaders}}.$$

On the other hand, we can first choose k leaders and then choose $j-k$ other members of the committee from the remaining $i-k$ students:

$$\#(\text{choices}) = \underbrace{\binom{i}{k}}_{\text{choose leaders}} \times \underbrace{\binom{i-k}{j-k}}_{\text{then choose committee}}.$$

Comparing these two gives the result. □

4.4 The Multinomial Theorem

In this section we will discuss a generalization of the binomial theorem.

We know that the number of words of length n from the alphabet $\{a, b, c\}$ is 3^n. Furthermore, we know that the number of these words containing i copies of the letter a and $n-i$ copies of the letter b is $\binom{n}{i}$. Indeed, we just need to choose i of the n positions in which to place the a's. Then all of the remaining letters are b's. More generally, how many words are there containing i copies of a and j copies of b?

Answer. First choose i out of n positions to place the a's. Then from the remaining $n-i$ positions choose j positions to place the b's. Thus we have

$$\#\left(\begin{array}{c}\text{words of length } n \text{ containing} \\ i \text{ copies of } a \text{ and } j \text{ copies of } b\end{array}\right) = \underbrace{\binom{n}{i}}_{\text{first place the } a\text{'s}} \times \underbrace{\binom{n-i}{j}}_{\text{then place the } b\text{'s}}.$$

□

These numbers are called *trinomial coefficients* because of the following algebraic formula:

$$(a+b+c)^n = \sum_{i,j,k} \frac{n!}{i!j!k!} a^i b^j c^k.$$

Does this formula simply? Sure:[54]

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

And what is the significance of the number $n-i-j$? Since there are only three letters this must be the number of c's n the word. Thus we can also write

$$\#\left(\begin{array}{c}\text{words made from } i \text{ copies of } a, \\ j \text{ copies of } b \text{ and } k \text{ copies of } c\end{array}\right) = \frac{n!}{i!j!k!} = \frac{(i+j+k)!}{i!j!k!}.$$

[54] You can also view this as a committee of size $i+j$ with

These numbers are called *trinomial coefficients* because of the following formula, which is called the *trinomial theorem*:

$$(a+b+c)^n = \sum_{i,j,k} \frac{n!}{i!j!k!} a^i b^j c^k.$$

Here we think of the symbols a, b, c as **numbers** that can be added and multiplied and we sum over all integers $i, j, k \geqslant 0$ such that $i + j + k = n$. For example, when $n = 2$ we have

$$(a+b+c)^2 = \begin{pmatrix} \frac{2!}{1!1!0!}a^1b^1c^0 & +\frac{2!}{1!1!0!}a^1b^1c^0 & +\frac{2!}{0!2!0!}a^0b^2c^0 \\ +\frac{2!}{1!0!1!}a^1b^0c^1 & +\frac{2!}{0!1!1!}a^0b^1c^1 & \\ +\frac{2!}{0!0!2!}a^0b^0c^2 & & \end{pmatrix} = \begin{pmatrix} a^2 & +2ab & +b^2 \\ +2ac & +2bc & \\ +c^2 & & \end{pmatrix}.$$

And why are there 6 terms in this expansion? Each term corresponds to a solution of the equation $i + j + k = 2$ where $i, j, k \in \mathbb{N}$. We saw in section 4.2 that the number of solutions is

$$\binom{\text{sum} + \#\text{ variables} - 1}{\text{sum}} = \binom{2+3-1}{2} = \frac{4}{2} = 6.$$

We can see what this has to do with counting **words** if we temporarily pretend that $ab \neq ba$, $ac \neq ca$ and $bc \neq cb$. Then the expansion becomes

$$\begin{aligned}(a+b+c)^2 &= aa + (ab+ba) + bb + (ac+ca) + (bc+cb) + cc \\ &= \text{the sum of all words of length 2 from the alphabet } \{a, b, c\}.\end{aligned}$$

Here is the general situation.

The Multinomial Theorem

Consider an alphabet $\{a_1, a_2, \ldots, a_\ell\}$ of size ℓ. Then the number of words of length n from this alphabet which contain k_i copies of the letter a_i is equal to the following number, called a *multinomial coefficient*:

$$\binom{n}{k_1, k_2, \ldots, k_\ell} := \frac{n!}{k_1!k_2!\cdots k_\ell!}.$$

If we temporarily treat the letters a_i as numbers then this fact is equivalent to the *multinomial theorem*:

$$(a_1 + a_2 + \cdots + a_\ell)^n = \sum_{k_1,\ldots,k_\ell} \binom{n}{k_1, \ldots, k_\ell} a_1^{k_1} \cdots a_\ell^{k_\ell}.$$

On the right we sum over all non-negative integers $k_1, \ldots, k_\ell$ such that $k_1 + \cdots + k_\ell = n$. The number of summands is thus $\binom{\ell+n-1}{n}$.

As with the binomial theorem, the multinomial theorem can be proved in many different ways. (One of these methods is by induction, which you will examine on the homework.) Instead of proving it now, let us examine some special cases.

Example: Alphabet of length $\ell = 2$. Consider the alphabet $\{a_1, a_2\}$. Then the number of words of length n containing k_1 copies of a_1 and k_2 copies of a_2 is

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

Since $k_1 + k_2 = n$ we could also write this as

$$\binom{n}{k_1, k_2} = \frac{n!}{k_1!(n-k_1)!} = \binom{n}{k_1}$$

or

$$\binom{n}{k_1, k_2} = \frac{n!}{(n-k_2)! k_2!} = \binom{n}{k_2}.$$

Indeed, to create a word of length n containing k_1 copies of a_1 and k_2 copies of a_2 we can either choose the positions for the a_1's in $\binom{n}{k_1}$ ways or choose the positions for the a_2's in $\binom{n}{k_2}$ ways. The result is the same.

The multinomial theorem in this case is just the binomial theorem in disguise:

$$(a_1 + a_2)^n = \sum_{\substack{k_1, k_2 \geqslant 0 \\ k_1 + k_2 = n}} \binom{n}{k_1, k_2} a_1^{k_1} a_2^{k_2} = \sum_{k=0}^{n} \binom{n}{k} a_1^k a_2^{n-k}.$$

Example: Rearrangements of Mississippi. In how many ways can you arrange the following letters?

$$m, i, s, s, i, s, s, i, p, p, i$$

We are looking for a word of length 11 from the alphabet $\{m, i, s, p\}$ in which the letter m appears once, the letter i appears 4 times, the letter s appears 4 times and the letter p appears twice. Thus the answer is

$$\binom{11}{1, 4, 4, 2} = \frac{11!}{1!4!4!2!} = 34650 \text{ ways.}$$

This is related to the multinomial theorem because the word *mississippi* is one of the terms in the expansion of $(m + i + s + p)^{11}$:

$$(m + i + s + p)^{11} = \cdots + mississippi + \cdots.$$

If we collect all of the terms with the same number of each letter then we get

$$(m+i+s+p)^{11} = \cdots + 34650 \cdot m^1 i^4 s^4 p^2 + \cdots .$$

There is another way to think of this problem if you like. Consider the **labeled** letters:

$$m_1, i_1, i_2, i_3, i_4, s_1, s_2, s_3, s_4, p_1, p_2.$$

Let S be the set of words that we can make with the labeled letters. Since each of the 11 letters is distinct we have

$$\#S = 11!.$$

On the other hand, we can form such a word by **first** choosing an unlabeled word and **then** placing labels on the letters. Let N be the number of words that we can make from the unlabeled letters. Then we obtain

$$\#S = \underbrace{N}_{\text{choose unlabeled word}} \times \underbrace{1!}_{\text{label the } m\text{'s}} \times \underbrace{4!}_{\text{label the } i\text{'s}} \times \underbrace{4!}_{\text{label the } s\text{'s}} \times \underbrace{2!}_{\text{label the } p\text{'s}} .$$

Finally, equating these two expressions for $\#S$ gives

$$N \cdot 1!4!4!2! = 11!$$
$$N = \frac{11!}{1!4!4!2!},$$

as expected. This method is an example of double counting.

4.5 Newton's Binomial Theorem

In this section we will discuss a **different** generalization of the binomial theorem. We have seen that the number of permutations (ordered selections without repetition) of k things chosen from n equals

$${}_nP_k = n(n-1)(n-2)\cdots(n-k+1).$$

Furthermore, we have seen that the number of combinations (unordered selections without repetition) of k things chosen from n is given by

$${}_nC_k = \frac{{}_nP_k}{k!}.$$

At first it seems that these formulas only make sense for integers $0 \leqslant k \leqslant n$. However, in this section we will observe that the formulas can be defined for **any number** n. To emphasize that n is no longer an integer, we will use a different notation.

Definition of Falling Factorials

Fix an integer $k \geqslant 0$. Then for **any number** z we define the *falling factorial*

$$(z)_k := \begin{cases} 1 & \text{if } k = 0, \\ z(z-1)(z-2)\cdots(z-k+1) & \text{if } k \geqslant 1. \end{cases}$$

We will also define the following notation:

$$\binom{z}{k} := \frac{(z)_k}{k!}.$$

It is not immediately clear whether this notation is useful.

We should make a few observations right away:

- If $0 \leqslant k \leqslant n$ are integers then we have

$$(n)_k = n(n-1)(n-2)\cdots(n-k+1) = \frac{n!}{(n-k)!}$$

  and hence $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ as usual.

- If $0 \leqslant n < k$ are integers then one of the factors in $(n)_k$ is equal to zero and hence $(n)_k = 0$. It follows also that $\binom{n}{k} = 0$, which makes sense I guess, since there are zero ways to choose k things from n things when $k > n$.

Thus the new situations will occur when z is a negative integer or a non-integer number. For practice, let us compute the value of $\binom{-1}{k}$:

$$\begin{aligned}
\binom{-1}{k} &= \frac{1}{k!} \cdot (-1)_k \\
&= \frac{1}{k!} \cdot (-1)(-1-1)(-1-2)\cdots(-1-k+1) \\
&= \frac{1}{k!} \cdot (-1)(-2)(-3)\cdots(-k) \\
&= \frac{1}{k!} \cdot (-1)^k k! \\
&= (-1)^k.
\end{aligned}$$

Well, okay. It makes no sense to "choose k things from -1 things", but the algebraic formula $\binom{-1}{k} = (-1)^k$ is still perfectly valid. We can understand this formula a bit better with the following theorem.

The Generalized Pascal's Triangle

For all integers $n, k \in \mathbb{Z}$ with $k \geqslant 0$ we have

$$\binom{n}{k} = \begin{cases} 1 & \text{if } k = 0, \\ 0 & \text{if } n = 0 \text{ and } k \neq 0, \\ \binom{n-1}{k-1} + \binom{n-1}{k} & \text{otherwise.} \end{cases}$$

Proof. The boundary cases are immediate. For the recurrence, we observe that the usual algebraic proof goes through even for **non-integer values of** n:

$$\begin{aligned} \binom{z-1}{k-1} + \binom{z-1}{k} &= \frac{(z-1)_{k-1}}{(k-1)!} + \frac{(z-1)_k}{k!} \\ &= \frac{k(z-1)_{k-1}}{k(k-1)!} + \frac{(z-1)_k}{k!} \\ &= \frac{(z-1)_{k-1}}{k!} + \frac{(z-1)_k}{k!} \\ &= \frac{1}{k!}\left[k(z-1)_{k-1} + (z-1)_k\right] \\ &= \frac{1}{k!}(z-1)_{k-1}\left[\cancel{k} + (z-\cancel{1}-\cancel{k}+\cancel{1})\right] \\ &= \frac{1}{k!}(z-1)_{k-1} \cdot z \\ &= \frac{1}{k!}(z)_k \\ &= \binom{z}{k}. \end{aligned}$$

□

Here is a picture of the generalized Pascal's triangle:

$$\begin{array}{ccccccccccccccc}
 & & & & & & & 1 & & -5 & & 15 & & & \\
 & & & & & & 1 & & -4 & & 10 & & \cdots & & \\
 & & & & & 1 & & -3 & & 6 & & -10 & & & \\
 & & & & 1 & & -2 & & 3 & & -4 & & \cdots & & \\
 & & & 1 & & -1 & & 1 & & -1 & & 1 & & & \\
 & & 1 & & 0 & & 0 & & 0 & & 0 & & \cdots & & \\
 & 1 & & 1 & & 0 & & 0 & & 0 & & 0 & & & \\
1 & & 2 & & 1 & & 0 & & 0 & & 0 & & \cdots & & \\
 & 3 & & 3 & & 1 & & 0 & & 0 & & 0 & & & \\
 & & 4 & & 6 & & 4 & & 1 & & 0 & & 0 & & \cdots
\end{array}$$

Do you notice anything? It seems that the numbers in the "negative rows" are just another copy of Pascal's triangle, but rotated and with alternating minus signs. On the homework you

will verify this observation by proving for all integers $n, k \geqslant 0$ that

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

Wait a minute. The number $\binom{n+k-1}{k}$ on the right side is equal to the number of ways to choose k things from n things when repetition is allowed. At first this seems like a strange coincidence but I will show you that there is a good reason for it.

The following generalization of the binomial theorem was discovered by Isaac Newton around 1665 and played an important role in his development of Calculus.

Newton's Binomial Theorem (1665)

Let x and z be any real or complex numbers with $|x| < 1$. Then we have the following identity, where the infinite series on the right is convergent:

$$(1+x)^z = \sum_{k=0}^{\infty} \binom{z}{k} x^k = \binom{z}{0} + \binom{z}{1} x + \binom{z}{2} x^2 + \cdots.$$

Note that this reduces to the usual binomial theorem when $z = n$ is a positive integer, since then we have $\binom{n}{k} = 0$ for all $k > n$. If z is not a positive integer then the series is truly infinite. We can compute the first few negative integer expansions by looking at the table above:

$$\begin{aligned}
(1+x)^{-1} &= 1 - x + x^2 - x^3 + x^4 - \cdots \\
(1+x)^{-2} &= 1 - 2x + 3x^2 - 4x^3 + \cdots \\
(1+x)^{-3} &= 1 - 3x + 6x^2 - 10x^3 + \cdots
\end{aligned}$$

Indeed, we can verify that these formulas are true because the first is just the geometric series:

$$\begin{aligned}
(1-x)^{-1} &= 1 + x + x^2 + x^3 + x^4 + \cdots \\
(1-(-x))^{-1} &= 1 + (-x) + (-x)^2 + (-x)^3 + (-x)^4 + \cdots \\
(1+x)^{-1} &= 1 - x + x^2 - x^3 + x^4 - \cdots.
\end{aligned}$$

The second formula is obtained by differentiating the geometric series as a function of x:

$$\begin{aligned}
\frac{d}{dx}(1+x)^{-1} &= \frac{d}{dx}(1 - x + x^2 - x^3 + x^4 - \cdots) \\
-(1+x)^{-2} &= 0 - 1 + 2x - 3x^2 + 4x^3 - \cdots \\
(1+x)^{-2} &= 1 - 2x + 3x^2 - 4x^3 + \cdots.
\end{aligned}$$

And the third formula is obtained by differentiating the second:

$$\begin{aligned}\frac{d}{dx}(1+x)^{-2} &= \frac{d}{dx}(1-2x+3x^2-4x^3+5x^4-\cdots)\\ -2(1+x)^{-3} &= 0-2+2\cdot 3\cdot x-3\cdot 4\cdot x^2+4\cdot 5\cdot x^3-\cdots\\ (1+x)^{-3} &= 1-\frac{2\cdot 3}{2}\cdot x+\frac{3\cdot 4}{2}\cdot x^2-\frac{4\cdot 5}{2}\cdot x^3+\cdots.\end{aligned}$$

You can probably see a pattern here. Indeed, we could use induction to prove Newton's formula for $(1+x)^{-n}$ when n is a positive integer. The interesting surprise that Newton discovered is that the formula holds also for non-integer exponents. In fact, one can use Newton's theorem and some tricky algebraic simplification to prove that

$$\sqrt{1+x} = (1+x)^{1/2} = \sum_{k=0}^{\infty}\binom{1/2}{k}x^k = \sum_{k=0}^{\infty}\frac{(-1)^{k-1}}{k\cdot 2^{2k-1}}\binom{2(k-1)}{k-1}x^k \quad \text{when } |x|<1.$$

Finally, let us return to the strange coincidence:

$$\begin{aligned}\binom{-n}{k} &= (-1)^k\binom{(n-1)+k}{k}\\ &= (-1)^k\left(\!\binom{n}{k}\!\right)\\ &= (-1)^k\cdot\#(\text{solutions } e_1,\ldots,e_n\in\mathbb{N} \text{ to the equation } e_1+\cdots+e_n=k).\end{aligned}$$

I claim that Newton's theorem explains this coincidence.

Generating Function for the Numbers $\left(\!\binom{n}{k}\!\right)$

For all integers $n \geqslant 0$ and complex numbers $|x|<1$ we have

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

We say that $1/(1-x)^n$ is a *generating function* for the numbers $\left(\!\binom{n}{k}\!\right)$ because it encodes these numbers as the coefficients in its power series expansion.

Of course we can give an algebraic proof.

Algebraic Proof. On the homework you will use algebraic manipulation to show that $\binom{-n}{k} = (-1)^k\left(\!\binom{n}{k}\!\right)$. Then from Newton's theorem we have

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

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

□

But this proof does not explain **why** the result is true. For that we need a counting proof.

Counting Proof. We can rewrite the formula using the geometric series:

$$\left(1 + x + x^2 + x^3 + \cdots\right)^n = \left(\frac{1}{1-x}\right)^n = \sum_{k=0}^{\infty} \left(\!\binom{n}{k}\!\right) x^k.$$

In order to prove that the formula is true we will compute the coefficient of x^k on each side. On the right side we have $\left(\!\binom{n}{k}\!\right)$. Now observe that every term on the left side has the form

$$x^{e_1} x^{e_2} \cdots x^{e_n} \quad \text{for some exponents } e_1, e_2, \ldots, e_n \in \mathbb{N}.$$

Indeed, to obtain a term on the left side we multiply one term from each copy of the geometric series. Since there are n copies, we end up with a product of n powers of x. Thus to compute the coefficient of x^k on the left we only need to count the number of choices of exponents $e_1, e_2, \ldots, e_n \in \mathbb{N}$ such that

$$x^k = x^{e_1} x^{e_2} \cdots x^{e_n} = x^{e_1+e_2+\cdots+e_n}.$$

In other words, we need to count the number of solutions $e_1, e_2, \ldots, e_n \in \mathbb{N}$ to the equation

$$e_1 + e_2 + \cdots + e_n = k.$$

From section 4.2 we know that the answer to this problem is $\left(\!\binom{n}{k}\!\right)$. □

Maybe you don't find that proof very convincing, because the multiplication of power series on the left happens mostly in our minds. To write it down explicitly would be a notational nightmare and would make the proof even less understandable. The reason I know that the proof is correct is because:

(1) It is plausible.

(2) We already know that the result is true.

That's good enough for me. The reason that the counting proof is important is because it explains that the coincidence $\binom{-n}{k} = (-1)^k \left(\!\binom{n}{k}\!\right)$ was not a coincidence.

In the next section we will investigate the idea of *generating functions* more systematically.

4.6 Generating Functions

Never mind. We don't have time for this.

If we had time we would discuss (1) binomial theorem, (2) fibonacci numbers, (3) integer partitions. Sometimes generating functions lead to the cleanest proofs.

4.7 Worked Exercises

4.1. The following table shows that the Fibonacci sequence can be run in both directions:

n	-4	-3	-2	-1	0	1	2	3	4
F_n	-3	2	-1	1	0	1	1	2	3

Use induction to prove that $F_{-n} = (-1)^{n+1}F_n$ for all $n \geqslant 0$.

Proof. The statement holds for $n = 0$ because $F_{-0} = F_0 = 0$ and $(-1)^{0+1}F_0 = 0$. Now assume for induction that $F_{-k} = (-1)^{k+1}F_k$ for all $k \in \{0, 1, \ldots, n-1\}$. Then we also have

$$\begin{aligned}
F_{-n+2} &= F_{-n+1} + F_{-n} && \text{definition} \\
F_{-n} &= F_{-n+2} - F_{-n+1} \\
&= F_{-(n-2)} - F_{-(n-1)} \\
&= (-1)^{(n-2)+1}F_{n-2} - (-1)^{(n-1)+1}F_{n-1} && \text{induction} \\
&= (-1)^{n+1}F_{n-2} + (-1)^{n+1}F_{n-1} \\
&= (-1)^{n+1}\left[F_{n-2} + F_{n-1}\right] \\
&= (-1)^{n+1}F_n && \text{definition}
\end{aligned}$$

□

4.2. For all integers $n \geqslant k > 0$ we have $k\binom{n}{k} = n\binom{n-1}{k-1}$.

(a) Prove this using pure algebra.

(b) Prove this using a counting argument. [Hint: Choose a committee of k people from n people. The committee has a president.]

(a) **Algebraic Proof.** The left side simplifies to

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

and the right side simplifies to the same expression:

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

□

(b) **Counting Proof.** Let N be the number of ways to choose a committee of k people from n people, where 1 member of the committee is the president. On the one hand, we can choose the commiittee first and then choose the president:

$$N = \underbrace{\binom{n}{k}}_{\text{choose the committee}} \times \underbrace{k}_{\text{then choose the president}} = k\binom{n}{k}.$$

On the other hand, we can choose one person from n to be the president and then choose $k-1$ from the remaining $n-1$ to be the other committee members:

$$N = \underbrace{n}_{\text{choose the president}} \times \underbrace{\binom{n-1}{k-1}}_{\text{then choose the other members}} = n\binom{n-1}{k-1}.$$

□

4.3. Count the possibilities in each case.

(a) A phone number consists of 7 digits.

(b) Suppose that a license plate consists of 3 digits followed by 4 letters.[55]

(c) A poker hand consists of 5 unordered cards from a standard deck of 52.

(d) Solutions $x_1, x_2, x_3, x_4, x_5 \in \mathbb{N}$ to the equation $x_1 + x_2 + x_3 + x_4 + x_5 = 10$.

(a) A phone number is a word of length 7 from the alphabet $\{0, 1, \ldots, 9\}$. Since repetition is allowed there are

$$10^7 = 10,000,000 \text{ different phone numbers.}$$

This is why we also have area codes.

(b) Symbols on a license plate are ordered. If symbols may be repeated then the number of possibilities is

$$\underbrace{10}_{\text{1st digit}} \times \underbrace{10}_{\text{2nd digit}} \times \underbrace{10}_{\text{3rd digit}} \times \underbrace{26}_{\text{1st letter}} \times \underbrace{26}_{\text{2nd letter}} \times \underbrace{26}_{\text{3rd letter}} \times \underbrace{26}_{\text{4th letter}} = 456,976,000.$$

If symbols may not be repeated then the number of possibilities is

$$\underbrace{10}_{\text{1st digit}} \times \underbrace{9}_{\text{2nd digit}} \times \underbrace{8}_{\text{3rd digit}} \times \underbrace{26}_{\text{1st letter}} \times \underbrace{25}_{\text{2nd letter}} \times \underbrace{24}_{\text{3rd letter}} \times \underbrace{23}_{\text{4th letter}} = 258,336,000.$$

[55] Assume that the alphabet has 26 letters.

This should be enough for most countries.

(c) Since a poker hand is unordered and cards are not repeated, the number of possibilities is

$$\binom{52}{5} = \frac{52 \cdot 51 \cdot 50 \cdot 49 \cdot 48}{5 \cdot 4 \cdot 3 \cdot 2 \cdot 1} = 2,598,960.$$

Will you ever see the same hand twice? According to my calculations, the average number of poker hands until you see the first repetition is 2021.17 (assuming that each hand comes from a full shuffled deck).[56]

(d) The number of solutions $x_1, \ldots, x_5 \in \mathbb{N}$ to the equation $x_1 + \cdots + x_5 = 10$ is the number of ways to buy 10 gallons of ice cream if there are 5 possible flavors (so x_i is the number of gallons of the ith flavor). We can encode such a choice as a binary sequence with ten 1's (representing the gallons of ice cream) and four 0's (representing walls between flavors). The number of such binary strings is

$$\binom{\text{total \# positions}}{\text{\# positions for 0's}} = \binom{10+4}{4} = \binom{14}{4} = \frac{14 \cdot 13 \cdot 12 \cdot 11}{4 \cdot 3 \cdot 2 \cdot 1} = 1001.$$

If you just want to memorize a formula: The number of solutions $x_1, \ldots, x_n \in \mathbb{N}$ to the equation $x_1 + \cdots + x_n = k$ is $\binom{n+k-1}{k} = \binom{n+k-1}{n-1}$. In our case $n = 5$ and $k = 10$.

4.4. For all integers $r, g, n \geqslant 0$ we have the following identity:

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

(a) Prove this identity. [Hint: There are r red balls and g green balls in an urn. You reach in and grab n balls (unordered and without repetition). Count the number of possibilities in two different ways.]

(b) Use the result of (a) to prove that $\binom{n}{0}^2 + \binom{n}{1}^2 + \binom{n}{2}^2 + \cdots + \binom{n}{n}^2 = \binom{2n}{n}$.

(a) **Counting Proof.** Let N be the number of ways to choose n balls from the urn. On the one hand, since there are $r + g$ balls in the urn we have

$$N = \binom{r+g}{n}.$$

[56] According to robjohn's answer on stackexchange, the expected number of rolls of a fair n-sided die until the first repetition is approximately $\sqrt{\frac{\pi n}{2}} + \frac{2}{3}$: `https://math.stackexchange.com/questions/542200/expected-number-of-tosses-before-you-see-a-repeat`

On the other hand, suppose that our selection of n balls contains exactly k red balls (and hence $n-k$ green balls). The number of ways this can happen is

$$\#\left(\begin{array}{c}\text{ways to get } k \text{ red balls}\\ \text{and } n-k \text{ green balls}\end{array}\right) = \underbrace{\binom{r}{k}}_{\text{choose red balls}} \times \underbrace{\binom{g}{n-k}}_{\text{then choose green balls}}.$$

Thus the total number of choices is

$$N = \sum_{k=0}^{n} \#\left(\begin{array}{c}\text{ways to get } k \text{ red balls}\\ \text{and } n-k \text{ green balls}\end{array}\right) = \sum_{k=0}^{n} \binom{r}{k}\binom{g}{n-k}.$$

If $r < n$ or $g < n$ then this sum contains some zeroes since we define $\binom{a}{b} = 0$ for $a < b$. □

Remark: This is called *Vandermonde's identity*. It is much harder to find an algebraic proof.

(b) Let $r = g = n$. Then since $\binom{n}{k} = \binom{n}{n-k}$ we obtain

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

4.5. The trinomial coefficient $\binom{n}{i,j,k} = \frac{n!}{i!j!k!}$ is the number of words of length n from the alphabet $\{a,b,c\}$ using i copies of a, j copies of b and k copies of c. These numbers satisfy the *trinomial recurrence*:

$$\binom{n}{i,j,k} = \binom{n-1}{i-1,j,k} + \binom{n-1}{i,j-1,k} + \binom{n-1}{i,j,k-1}.$$

(a) Prove the trinomial recurrence using pure algebra.

(b) Prove the trinomial recurrence using a counting argument.

(a) **Algebraic Proof.** We add three fractions by obtaining the common denominator $i!j!k!$:

$$\begin{aligned}
&\frac{(n-1)!}{(i-1)!j!k!} + \frac{(n-1)!}{i!(j-1)!k!} + \frac{(n-1)!}{i!j!(k-1)!}\\
&= \frac{i(n-1)!}{i(i-1)!j!k!} + \frac{j(n-1)!}{i!j(j-1)!k!} + \frac{k(n-1)!}{i!j!k(k-1)!}\\
&= \frac{i(n-1)!}{i!j!k!} + \frac{j(n-1)!}{i!j!k!} + \frac{k(n-1)!}{i!j!k!}\\
&= \frac{i(n-1)! + j(n-1)! + k(n-1)!}{i!j!k!}\\
&= \frac{(i+j+k)(n-1)!}{i!j!k!}
\end{aligned}$$

$$= \frac{n(n-1)!}{i!j!k!}$$
$$= \frac{n!}{i!j!k!}.$$

□

(b) **Counting Proof.** Recall that $\binom{n}{i,j,k}$ is the number of words of length n from the alphabet $\{a, b, c\}$ which contain i copies of a, j copies of b and k copies of c. Let S be the set of such words. Furthermore, let $S_a, S_b, S_c \subseteq S$ be the subsets of words in which a, b or c is the **first letter**, respectively. In order to choose an element of S_a we start by placing a on the left. Then we must fill in the remaining $n-1$ letters with a word of length $n-1$ using $i-1$ copies of a, j copies of b and k copies of c. The number of ways to do this is

$$\#S_a = \binom{n-1}{i-1, j, k}.$$

Similarly we have

$$\#S_b = \binom{n-1}{i, j-1, k} \qquad \text{and} \qquad \#S_c = \binom{n-1}{i, j, k-1}.$$

Since the sets S_a, S_b, S_c partition the set S we conclude that

$$\#S = \#S_a + \#S_b + \#S_c$$
$$\binom{n}{i,j,k} = \binom{n-1}{i-1,j,k} + \binom{n-1}{i,j-1,k} + \binom{n-1}{i,j,k-1}.$$

□

4.6. Let $k \geqslant 0$ be an integer. Then for any number z the following formula makes sense:

$$\binom{z}{k} := \frac{1}{k!} \cdot z(z-1)(z-2)\cdots(z-k+1).$$

Isaac Newton proved that for all numbers z, x with $|x| < 1$ the following series converges:

$$(1+x)^z = \binom{z}{0} + \binom{z}{1}x + \binom{z}{2}x^2 + \binom{z}{3}x^3 + \cdots.$$

(a) For all integers $n, k \geqslant 0$ show that $\binom{-n}{k} = (-1)^k\binom{n+k-1}{k}$.

(b) Use part (a) to obtain the power series expansion of $(1+x)^{-2}$.

(a) For all integers $n, k \geqslant 0$ we have

$$\binom{-n}{k} = \frac{1}{k!}(-n)(-n-1)(-n-2)\cdots(-n-k+1)$$

$$
\begin{aligned}
&= \frac{1}{k!}(-1)(n)(-1)(n+1)(-1)(n+2)\cdots(-1)(n+k-1) \\
&= \frac{1}{k!}(-1)^k(n+k-1)\cdots(n+2)(n+1)(n) \\
&= \frac{1}{k!}(-1)^k(n+k-1)\cdots(n+2)(n+1)(n)\frac{(n-1)!}{(n-1)!} \\
&= \frac{1}{k!}(-1)^k\frac{(n+k-1)!}{(n-1)!} \\
&= (-1)^k\frac{(n+k-1)!}{k!(n-1)!} \\
&= (-1)^k\binom{n+k-1}{k}.
\end{aligned}
$$

□

(b) For all integers $k \geqslant 0$, part (a) tells us that

$$\binom{-2}{k} = (-1)^k\binom{2+k-1}{k} = (-1)^k\binom{k+1}{k} = (-1)^k(k+1).$$

And then Newton's theorem tells us that

$$(1+x)^{-2} = \sum_{k=0}^{\infty}\binom{-2}{k}x^k = \sum_{k=0}^{\infty}(-1)^k(k+1)x^k = 1 - 2x + 3x^2 - 4x^3 + \cdots.$$

5 Graph Theory

The final topic of this course is graph theory. This is a relatively modern subject that one could describe as "the study of how things are connected".[57] It has applications to computer science at all levels. For example, it deals with the following problems:

- Given a logical circuit, how can we etch it onto a chip with the fewest number of wire crossings?
- What is the least number of wires that we need to connect n computers? What if each wire has an associated cost/distance?
- How can we design a network of computers that remains connected if any k of the computers/connections are knocked out?

We will see that graph theory arguments are similar to counting arguments, in that they cannot easily be described using algebra. Instead we have to rely on pictures and abstract reasoning. This is why it is the final topic of the course.

[57] More precisely, graph theory is the study of how **one-dimensional things** (such as wires) are connected. *Topology* is the study of how higher-dimensional things (such as surfaces) are connected.

5.1 Definitions and Degrees

A graph consists of a set of vertices (or nodes) that are connected by a set of edges. We can formalize this idea as follows.

Definition of Simple Graphs

A *simple graph* $G = (V, E)$ consists of:

- a set V of *vertices*,
- a set $E \subseteq \binom{V}{2}$ of *edges*.

Here we use the notation $\binom{V}{2}$ to denote the **set of two-element subsets of** V. Thus an edge $\{u, v\}$ is just an unordered pair of distinct vertices $u \neq v \in V$.[58]

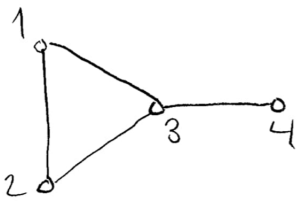
We encode the graph in a computer via the *adjacency matrix* $A = (a_{ij})$. If the vertices are ordered as $V = \{v_1, \ldots, v_n\}$ then the ij entry of the matrix is defined by

$$a_{ij} := \begin{cases} 1 & \text{if } \{v_i, v_j\} \in E, \\ 0 & \text{if } \{v_i, v_j\} \notin E. \end{cases}$$

For example, the following graph $G = (V, E)$ has

$$V = \{1, 2, 3, 4\} \qquad \text{and} \qquad E = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{3, 4\}.$$

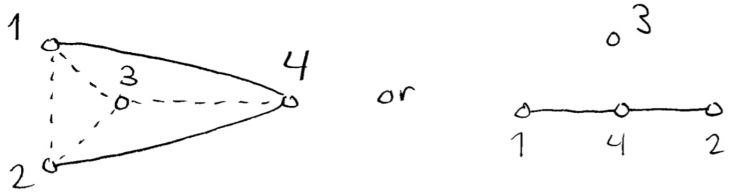
The adjacency matrix A is displayed on the right:



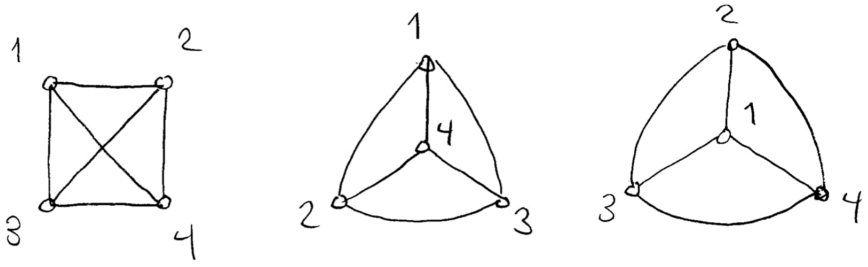
$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

Note that a graph with n vertices can have at most $\binom{n}{2}$ edges, since this is the number of unordered pairs of vertices. We can define the *complement* $G' = (V, E')$ of a graph $G = (V, E)$ by keeping the vertices the same and switching the edges with non-edges. For example, if G is the graph above then the complement G' has $V = \{1, 2, 3, 4\}$ and $E' = \{\{1, 4\}, \{2, 4\}\}$:

[58] There are other kinds of graphs. For example, *directed graphs* have edges corresponding to **ordered pairs** (u, v) of vertices. *Multigraphs* may have repeated edges $(\{u, v\}, \{u, v\})$ and/or loops (u, u). In this course we sill stick to simple graphs.



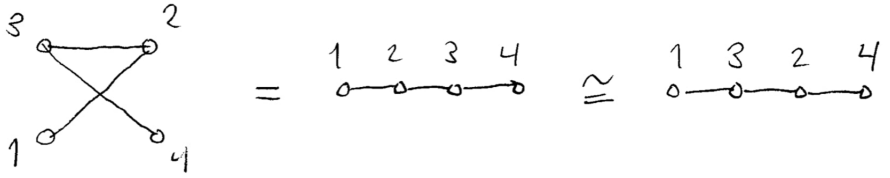
The data of a graph just tells us how the vertices are connected to each other. We can draw many different pictures to display this information. For example, let K_4 be the *complete graph* on the vertex set $\{1, 2, 3, 4\}$, which contains all six possible edges. Here are three different pictures of K_4:



Sometimes it is difficult to know whether two different pictures represent the same graph. In general, what do we mean when we say that two graphs are the same?

Definition of Graph Isomorphism

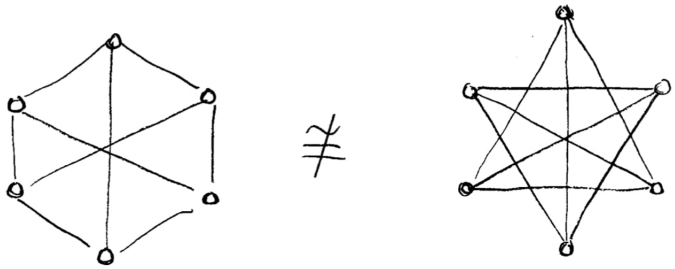
Consider two graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$. We say that G and H are *equal* (and write $G = H$) if $V_G = V_H$ and $E_G = E_H$. We say that G and H are *isomorphic* (and write $G \cong H$) if we have a bijection $V_G \leftrightarrow V_H$ that preserves the edges. For example:



If we only care about isomorphism then we will not label the vertices.

Sometimes it is difficult to tell whether two graphs are isomorphic. For example, you will

prove on the homework that the following two graphs are not isomorphic:

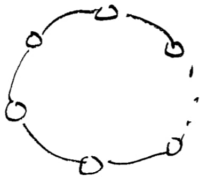


Don't worry, I'll give you a hint later. For now you can just puzzle over it. Here are some important classes of graphs that we use in our examples below.

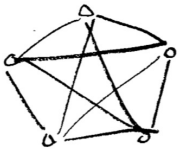
Important Clases of Graphs

- The *path* P_n on n vertices looks like this:

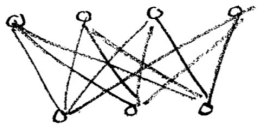
- The *cycle* C_n on n vertices looks like this:



- The *complete graph* K_n on n vertices has all $\binom{n}{2}$ possible edges. Here is K_5:



- The *complete bipartite graph* $K_{m,n}$ has vertex set $V = A \cup B$ with $\#A = m$ and $\#B = n$. The edge set is $\{\{a, b\} : a \in A \text{ and } b \in B\}$. Here is $K_{3,4}$:



Later I will define "bipartite graphs" more generally.

So that this first section is not **only** definitions, let me present the Handshaking Lemma and then show you a couple of quick applications.

The Handshaking Lemma

Let $G = (V, E)$ be a graph. For each vertex $u \in V$ we define its *degree* as follows:

$$\deg(u) := \#\{v \in V : \{u, v\} \in E\} = \#(\text{vertices sharing an edge with } u)$$

Then the sum of the degrees is equal to twice the number of edges:

$$\sum_{u \in V} \deg(u) = 2 \cdot \#E$$

(sum of the degrees) = (twice the number of edges).

Proof by Double Counting. Define a "lollipop" as an edge together with one of its vertices and let L be the set of lollipops in the graph. We will count this set in two different ways. On the one hand, we can choose the edge first, then choose the vertex:

$$\#L = \underbrace{\#E}_{\text{choose the edge}} \times \underbrace{2}_{\text{then choose the vertex}} = 2 \cdot \#E.$$

On the other hand, for each vertex $u \in V$ we note that there are $\deg(u)$ lollipops containing this vertex. Then summing over all vertices gives

$$\#L = \sum_{u \in V} \#\{\text{lollipops containing } u\} = \sum_{u \in V} \deg(u).$$

□

We have the following immediate corollary.

The Number of Odd Vertices is Even.

For any graph, the number of odd-degree vertices is even.

Proof. Let $V = V_e \cup V_o$, where V_e, V_o are the vertices of even and odd degree, respectively. Now assume for contradiction that $\#V_o$ is odd. Summing degrees for each set gives

$$\sum_{u \in V_o} \deg(u) = \sum (\text{odd number of odd numbers}) = (\text{odd number}),$$
$$\sum_{u \in V_e} \deg(u) = \sum (\text{some number of even numbers}) = (\text{even number}).$$

But then from the Handshaking Lemma we have

$$\sum_{u \in V} \deg(u) = \sum_{u \in V_e} \deg(u) + \sum_{u \in V_o} \deg(u)$$
$$2 \cdot \#E = \sum_{u \in V_e} \deg(u) + \sum_{u \in V_o} \deg(u)$$
$$(\text{even number}) = (\text{even number}) + (\text{odd number}),$$

which is a contradiction. □

Many nice graphs have the property that every vertex has the same degree. The following theorem tells us exactly when these graphs can exist.

Existence of Regular Graphs

We say that a graph is *d-regular* if each vertex has degree d.

(1) If there exists a d-regular graph on n vertices then dn is even.

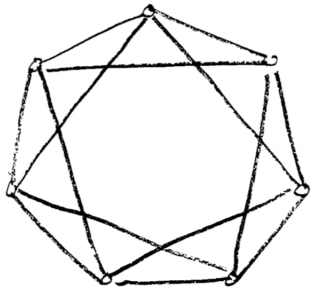
(2) If dn is even then there exists at least one d-regular graph on n vertices.

Proof. (1) Let G be a d-regular graph on n vertices. Then we have

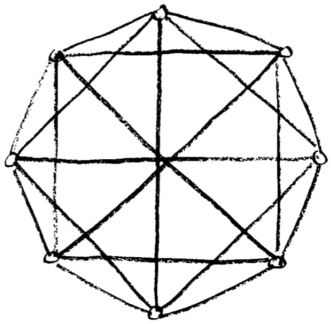
$$2 \cdot \#E = \sum_{u \in V} \deg(u) = \underbrace{d + d + \cdots + d}_{n \text{ times}} = dn.$$

(2) Conversely, suppose that dn is even. Then there are two cases. In each case we will construct a certain d-regular graph on n vertices, called a *Harary graph* $H_{d,n}$.

- If d is even then we arrange the n vertices in a circle and then connect each vertex to its $d/2$ neighbors on either side. For example, here is the graph $H_{4,7}$:



- If d is odd (hence n is even) then we draw n vertices in a circle and connect each vertex to its $(d-1)/2$ neighbors on either side. Furthermore, we connect each vertex to the opposite vertex (which exists because n is even). For example, here is the graph $H_{5,8}$:



□

Of course, there may exist many non-isomorphic d-regular graphs of size n. You have already seen two such graphs with $d = 3$ and $n = 6$. In general, it is a theorem of Béla Bollobás[59] that the number of non-isomorphic d-regular graphs on n vertices is approximately equal to

$$\frac{e^{-(d^2-1)/4}}{n!(d!)^n} \cdot \frac{(2m)!}{2^m m!}, \qquad \text{where } m = \frac{dn}{2} \text{ is an integer.}$$

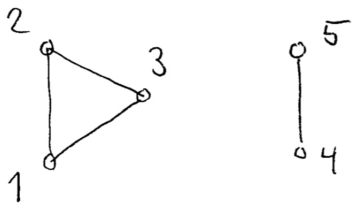
5.2 Paths and Components

What does it mean for a graph to be connected? For example, the graph defined by

$$V = \{1, 2, 3, 4, 5\} \quad \text{and} \quad E = \{\{1, 2\}, \{2, 3\}, \{1, 3\}, \{4, 5\}\}$$

is **not connected**:

[59] *The asymptotic number of unlabelled regular graphs*, (1981).



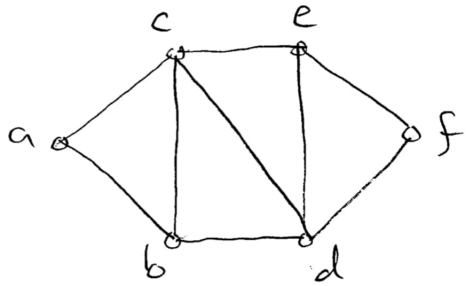
In fact, I would say that this graph has **two connected components**. In order to prove anything about connectivity we need a precise definition.

Definition of Walks and Paths

Let $G = (V, E)$ be a graph and consider two vertices $u, v \in V$.

- A *u, v-walk of length k*[60] is a sequence of vertices $u = v_0, v_1, \ldots, v_k = v$ in which $\{v_{i-1}, v_i\} \in E$ for all i. Note that a walk may contain repeated vertices and edges.
- A *u, v-path* is a u, v-walk with no repeated vertices. This implies that a path also has no repeated edges.

For example, consider the following graph on the vertex set $V = \{a, b, c, d, e, f\}$:

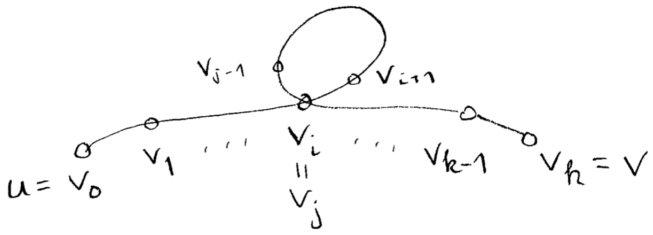


The sequence a, b, c, d, e, f is an a, f-path. The sequence a, b, d, e, c, d, f is an a, f-walk with the vertex d repeated. The sequence a, b, c, d, e, c, d, f is an a, f-walk with repeated edges and vertices.

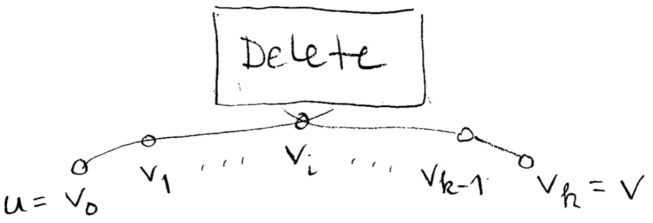
Note that walks can be very inefficient. It seems obvious that every u, v-walk can be shortened to a u, v-path by removing repetition, but this is a bit tricky to prove.

[60] We say it has length k because it travels across k edges.

Proof by Induction on Length. I claim that every u, v-walk of length $k \geqslant 1$ contains a u, v-path. For the base case we note that every walk of length 1 is a path, hence it contains a path (itself). Now consider a walk $u = v_0, v_1, \ldots, v_k = v$ of length k. If this walk is a path then we are done. Otherwise, we have a repeated vertex $v_i = v_j$ for some $i < j$:



Delete all steps between the two occurrences of v_i to obtain a shorter walk:



By (strong) induction on length, we may suppose that this shorter walk contains a u, v-path. But then this path is also contained in the original walk. [Note that the pictures are merely suggestive. There may be other repetitions that I have not drawn.] □

We can also express this proof as an algorithm: Find a repeated vertex. Delete everything between the two occurrences of this vertex. Continue until there are no repeated vertices. Since the length goes down at each step, the well-ordering principle guarantees that this process will terminate.

Now I can give the official definition of connectivity.

Definition of Connected Components

Let $G = (V, E)$ be a graph. For any two vertices $u, v \in V$ we define the following relation:

$$u \sim v \quad \Longleftrightarrow \quad \text{there exists a } u, v\text{-path.}$$

In this case we say that u and v are *connected*. We say that the graph G is *connected* if we have $u \sim v$ for all $u, v \in V$.

If G is not connected then the relation $\sim$ allows us to partition the vertices into *connected components*

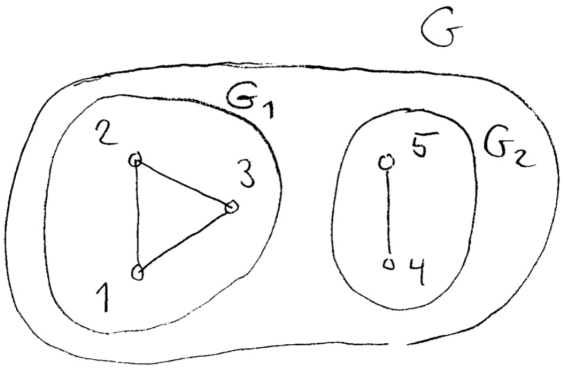
$$V = V_1 \cup V_2 \cup \cdots \cup V_k$$

so that $u \sim v$ if and only if u and v are in the same component. Let G_i denote the graph with vertices V_i and with edges inherited from G. Then we will write

$$G = G_1 \cup G_2 \cup \cdots \cup G_k$$

and we will call these the *connected components* of G. These are sometimes also called the *path components* of G.

For example, the graph from the beginning of this section decomposes into two connected components, $G = G_1 \cup G_2$:



Let me show you a cute and surprising result that we can prove by combining this definition with the Handshaking Lemma.

Cute and Surprising Result

If a graph has exactly two odd vertices then there exists a path between these vertices.

Non-Constructive Proof. Let $G = (V, E)$ and let $u, v \in V$ be the two vertices of odd degree. Now assume for contradiction that there does not exist a u, v-path. This means that u and v are in separate path components, say $u \in G_u$ and $v \in G_v$. But then G_u (and also G_v) is a graph with exactly one odd vertex, which contradicts the Handshaking Lemma. ▫

This proof is called "non-constructive" because it only proves the existence of a path; it does not tell us how to find a path. Here is a constructive proof, which, however, is not cute.

Constructive Proof. Let $G = (V, E)$ be a graph with exactly two odd vertices $u, v \in V$. I claim that the following algorithm produces a u, v-walk (which then contains a u, v-path):

procedure: to construct u, v-walk
$v_0 := u$
$v_1 :=$ any neighbor of v_0
while $v_k \neq v$ **do**
  $k := k + 1$
  $v_k :=$ any neighbor of v_{k-1}
  delete one copy of the edge $\{v_{k-1}, v_k\}$ from the graph[61]
end do
return $(v_0, v_1, v_2, \ldots, v_k)$

Suppose at the kth step that we have $v_k \neq v$. Then I claim that v_k still has a neighbor. Indeed, if $v_k = u$ then we have deleted an even number of edges at u. Since $\deg(u)$ is odd this means that u still has a neighbor. And if $v_k \notin \{u, v\}$ then we have deleted an odd number of edges at v_k. Since $\deg(v_k)$ is even this means that v_k still has a neighbor. Thus the algorithm never gets stuck. Finally, we observe that the algorithm must terminate because the graph has a finite number of edges. □

In the rest of this section we will prove a more substantial result that relates the number of edges to the number of connected components of a graph.

Connected Components Theorem

Let G be a graph[62] with n vertices, e edges and k connected components. Then we have

$$(n - k) \leqslant e \leqslant \binom{n - k + 1}{2} = \frac{(n - k + 1)(n - k)}{2}.$$

For the first inequality we observe that deleting an edge increases the number of components by 0 or 1.[63] The only difficulty is to turn this into an induction proof.

Proof that $n - k \leqslant e$. We will prove this by induction on the number of edges. First note that $e = 0$ implies $n = k$ (because each component is an isolated vertex), hence $n - k = 0 \leqslant 0 = e$.

[61] This proof works also for multigraphs.
[62] The upper bound holds for simple graphs. The lower bound holds for any kind of graph.
[63] See the definition of "bridges" in Section 5.4 for a proof of this observation.

Now consider a graph G with parameters n, e, k where $e \geqslant 1$. Delete a random edge to obtain a new graph G' with parameters n', e', k'. Observe that $n' = n$ and $e' = e - 1$. What about k? By deleting an edge we either preserve the number of components or we split one component into two. In either case we have $k' \leqslant k + 1$. Now since $e' < e$ we may assume by induction that $e' \geqslant n' - k'$ and hence

$$e - 1 = e' \geqslant n' - k' \geqslant n' - (k + 1) = n - (k + 1) = (n - k) - 1.$$

Adding 1 to both sides gives the result. □

The second inequality is trickier.

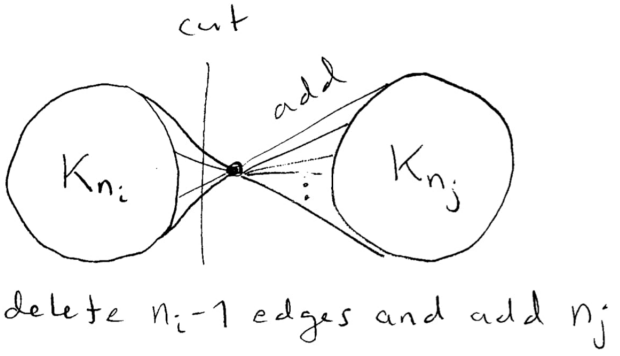
Proof that $e \leqslant \binom{n-k+1}{2}$. Let G be a graph with n vertices and k connected components $G_1, G_2, \ldots, G_k$. In this case we will prove that $e \leqslant \binom{n-k+1}{2}$. Suppose that the i-th component G_i has n_i edges. By adding all possible edges we observe that the number of edges is maximized when each component is a complete graph K_{n_i} with $\binom{n_i}{2}$ edges. In other words, we must have

$$e = \binom{n_1}{2} + \binom{n_2}{2} + \cdots + \binom{n_k}{2}.$$

It only remains to choose the values of $n_1, \ldots, n_k$ so that this number is **maximized**. I claim that this happens when all but one of the components are isolated vertices, i.e., when $n_i = 1$ for $k - 1$ of the values of i and hence hence $n_i = n - (k - 1)$ for the final value of i. In this case we will have

$$e = \binom{1}{2} + \binom{1}{2} + \cdots + \binom{1}{2} + \binom{n - (k-1)}{2} = 0 + \cdots + 0 + \binom{n-k+1}{2}$$

as desired. So let us assume for contradiction that the number e is maximized and that there exist two components $G_i \cong K_{n_i}$ and $G_j \cong K_{n_j}$ with $2 \leqslant n_i \leqslant n_j$. Let us disconnect one vertex from G_i and connect it to all of the vertices in G_j, which has the effect of replacing the numbers (n_i, n_j) by $(n_i - 1, n_j + 1)$. In doing so we will delete $n_i - 1$ edges from the graph and add n_j new edges to the graph:



Since $n_i \geqslant n_j$ this will **increase the number of edges**. But this is a contradiction because we assumed that e was maximized. ▫

This is a very typical graph theory proof since the essential insight is a picture: disconnect a vertex from one complete component and glue it to a larger component. The hard part is turning the picture into a convincing proof. There are no rules for doing this; the only criterion is that your audience believes that the proof is correct.

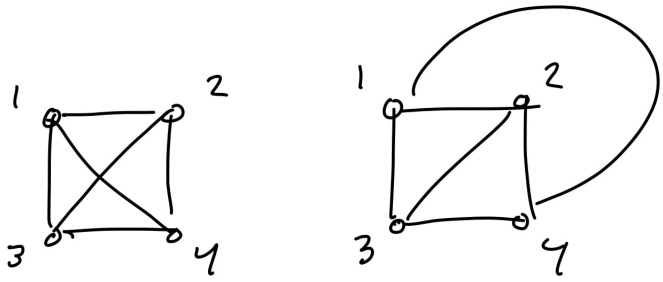
You will use the Connected Components Theorem on the homework to prove that any (simple) graph on n vertices with more than $\binom{n-1}{2}$ edges must be connected. More generally, one can show that any graph on n vertices with more than $\binom{n-r+1}{2}$ edges must have **fewer than** r **connected components**.

5.3 Planar Graphs

Let G be the graph with adjacency matrix

$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

This graph can be drawn in various ways. Here are two examples:

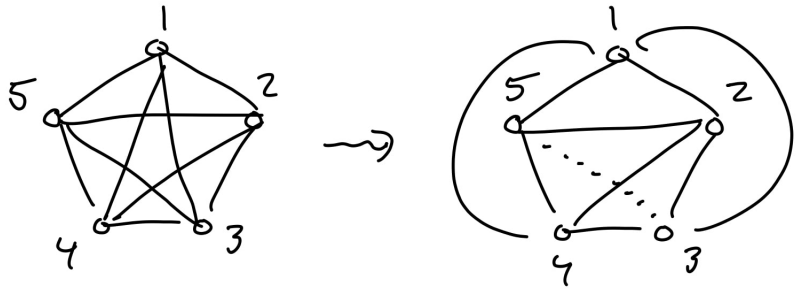


The drawing on the right has the nice property that the edges do not cross.

Definition of Planar Graphs

Inside a computer, a graph G is just a collection of information saying which vertices are connected by edges. As humans, we like to draw pictures of graphs. We say that G is a *planar graph* if it can be drawn in the plane in such a way that the edges do not cross.

For example, we just saw that the complete graph K_4 is planar. Is every graph planar? Consider the complete graph K_5. Here is an attempt to draw K_5 without edge crossings:



I moved the edges 13 and 14 outside of the main pentagon, but now I can't figure out how to draw the edge between vertices 3 and 5. After some trial and error you will become convinced that the graph K_5 is **not planar**. But how can we possibly prove this? In order to prove that a graph is planar we only have to produce a drawing. In order to prove that a graph is not planar, we need to show that all of the infinitely many possible drawings must have edge crossings. That seems hard.

The following theorem is our main tool for studying planar graphs.

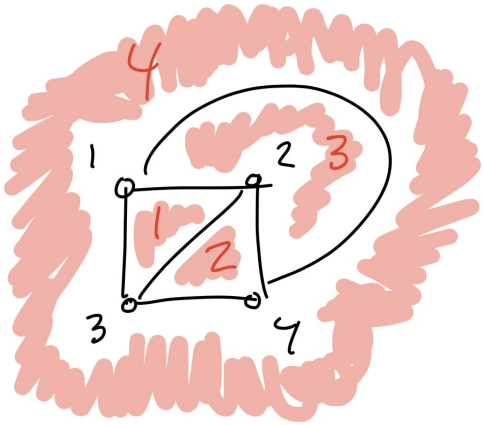
Euler's Formula for Planar Graphs

Let G be a connected planar graph with n vertices and e edges. Any planar drawing of G will divide the plane into regions, called *faces*. Note that there will be one "infinite face" on the outside of the drawing.[64] Let f be the number of faces in such a drawing, including the infinite face. Then we must have

$$n - e + f = 2.$$

For example, our planar drawing of K_4 has 4 faces:

[64] It is more natural to consider drawings on the surface of a sphere. Then all of the faces are finite.

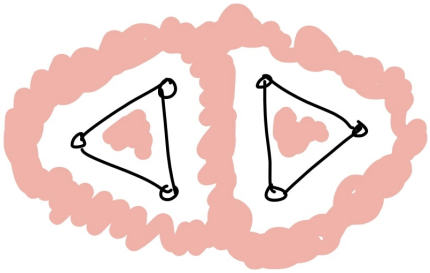


Since K_4 has $n = 4$ vertices and $e = 6$ edges, we confirm that

$$n - e + f = 4 - 6 + 4 = 2.$$

The numbers n and e are intrinsic properties of the graph, but the number f is a property of the drawing. One consequence of Euler's formula is that any two drawings of a planar graph must have the same number of faces: $f = 2 - n + e$.

Note that Euler's formula only applies to connected graphs. For example, here is a planar drawing of the disconnected graph $K_3 \cup K_3$:



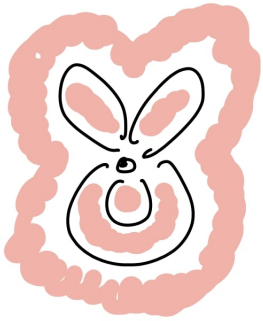
Note that this planar drawing has $n = 6$, $e = 6$ and $f = 3$, so that $n - e + f = 3$. Oops. If G is a planar graph with n vertices, e edges and k connected components, then a generalization of Euler's formula says that

$$v - e + f = k + 1.$$

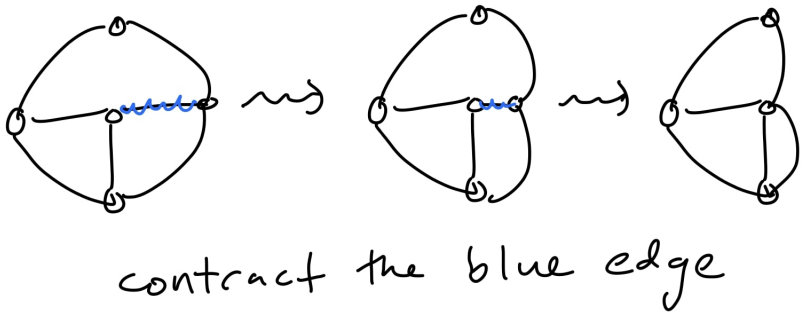
This is an easy consequence of the connected case, but we won't bother to prove it.

Proof of Euler's Formula. In order to make the proof easier, we will allow connected planar graphs with loops and multiple edges. We use induction on the number of vertices n.

For the base case we have $n = 1$. A planar graph with $n = 1$ vertex can have any number of loops e. If $e = 0$ then there is just one face (the infinite face). Each loop added to this will create one new face, so that $f = e + 1$. Then it follows that $n - e + f = 1 - e + (e - 1) = 2$, as desired. Here is an example with $n = 1$, $e = 4$ and $f = 5$:



Now fix some $n \geqslant 2$ and consider a planar drawing with n vertices, e edges and f faces.[65] Since our graph G is connected it must have an edge. Pick a random edge and contract it to a point, to obtain a new graph G'. Here is an example of an edge contraction:



Note that an edge contraction can create multiple edges and loops, which is why we have to include them in our proof. Let n', e' and f' be the number of vertices, edge and faces in the new graph G'. We observe that

$$n' = n - 1, \quad \text{and} \quad e' = e - 1 \quad \text{and } f' = f.$$

Indeed, contracting an edge removes one vertex and one edge, but it **does not change the number of faces**.[66] By induction on n, we may assume that $n' - e' + f' = 2$. But then we also have

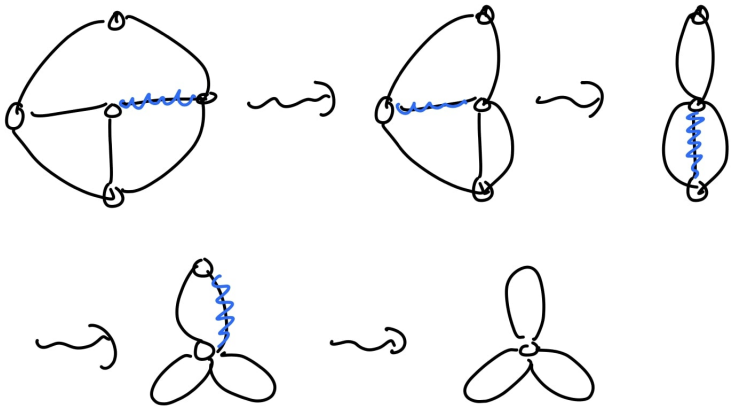
$$n - e + f = (n' + 1) - (e' + 1) - f' = n' - e' + f' = 2,$$

as desired. □

The induction proof stopped after one edge contraction. In the real life situation we would continue to contract random edges until we reach a graph with one vertex:

[65] This is a funny situation. We have to consider a specific drawing, even though we don't know what it looks like.

[66] The fact that the number of faces is unchanged is intuitively obvious but it is difficult to prove rigorously. Look at the *Jordan curve theorem* on Wikipedia. Rigorous topological proofs are one of my least favorite kinds of mathematics; it's just so much work for something that we already knew.



At each step, the quantity $n - e + f$ is unchanged. Since the final graph (with $n = 1$) satisfies $n - e + f = 2$, so must the original graph.

Now we will use Euler's formula to prove that K_5 is not planar. In order to facilitate the proof we give a version of the Handshaking Lemma for faces in a planar graph.

Handshaking for Faces in a Planar Graph

Consider a planar drawing of a graph $G = (V, E)$ and let F be the set of faces. For each face $f \in F$ we define its *degree*:[67]

$$\deg(f) := \text{the number of edges bounding the face } f.$$

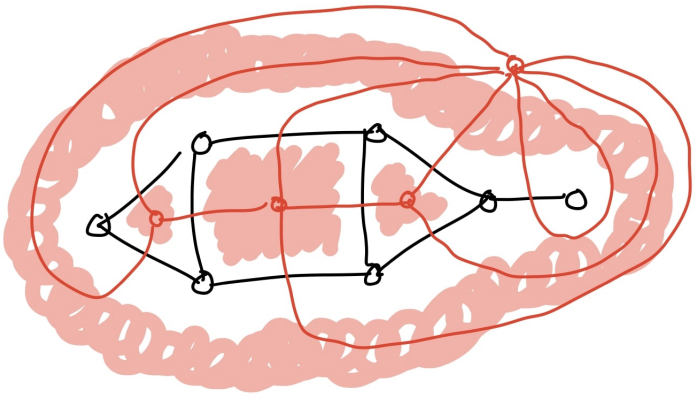
I claim that

$$\sum_{f \in F} \deg(f) = 2 \cdot \#E.$$

Technicality: If an edge e has the same face f on both sides, then we must count e twice in the degree of f. We have to do this to make the formula work.

We can prove this from scratch or we can derive it from the usual Handshaking Lemma for vertex degrees. To prove it from scratch we note that each edge gets counted twice in the sum of the face degrees. To derive it from the theorem for vertices, we draw the *dual planar graph*. For this, we put a vertex in each face and we draw an edge across each edge shared by the corresponding faces. That's hard to say and easier to draw:

[67] I apologize for using the letter f here for a face when when previously I used it for the number of faces. I couldn't think of a better notation.



A planar graph and its dual have the same number of edges, but the numbers of vertices and faces have been swapped. Furthermore, the degree of a vertex in the dual graph equals the degree of the corresponding face in the original graph. In the above example, we have $n = 7$, $e = 9$ and $f = 4$. The degrees of the faces are $3, 3, 4, 8$ and the sum of the face degrees is

$$\sum_{f \in F} \deg(f) = 3 + 3 + 4 + 8 = 18 = 2 \cdot 6 = 2e,$$

as expected.

Now we combine Euler's formula and Handshaking to derive an inequality relating the numbers of vertices and edges in a planar graph.

Vertex-Edge Inequalities for Planar Graphs

Let G be a connected graph with n vertices and e edges. Let $\ell \geqslant 3$ and suppose that G contains no cycles of length $< \ell$.[68] If G is planar then we must have

$$e \leqslant \frac{\ell(n-2)}{\ell - 2}.$$

For example, if G is a simple graph then we can take $\ell = 3$ because a simple graph has no loops (so no cycles of length 1) and no multiple edges (so no cycles of length 2). Thus a simple, connected planar graph must satisfy

$$e \leqslant \frac{3(n-2)}{3-2} = 3n - 6.$$

[68] A cycle is just a path that starts and ends at the same vertex. For the official definition see the next section.

Why did I bother to take ℓ as a variable, when I could have just assumed $\ell = 3$? See the homework.

Proof. Let F be the set of faces in a planar drawing of a connected graph $G = (V, E)$. If G contains no cycles of length $< \ell$ then we must have $\deg(\varphi) \geqslant \ell$ for all faces $\varphi \in F$.[69] Indeed, the edges surrounding a face φ are a cycle of length $\deg(\varphi)$. Then from Handshaking we have

$$2e = \sum_{\varphi \in F} \deg(\varphi) \geqslant \underbrace{\ell + \ell + \cdots + \ell}_{f \text{ times}} = \ell f.$$

Now combine this with Euler's formula $n - e + f = 2$ to obtain

$$\begin{aligned} \ell f &\leqslant 2e \\ \ell(2 - n + e) &\leqslant 2e \\ 2\ell - \ell n + \ell e &\leqslant 2e \\ \ell e - 2e &\leqslant \ell n - 2\ell \\ (\ell - 2)e &\leqslant \ell(n - 2) \\ e &\leqslant \ell(n - 2)/(\ell - 2). \end{aligned}$$

□

Finally, we can prove that K_5 is not planar.

Proof that K_5 is not planar. Assume for contradiction that K_5 has a planar drawing. Since K_5 is a simple graph (no loops and no multiple edges), the previous result says that

$$e \leqslant 3n - 6.$$

But K_5 has $n = 5$ vertices and $e = \binom{5}{2} = 10$ edges, which contradicts this inequality. □

More generally, the complete graph K_n has n vertices and $e = \binom{n}{2} = n(n-1)/2$ edges. If K_n is planar then we must have

$$\begin{aligned} e &\leqslant 3n - 6 \\ n(n-1)/2 &\leqslant 3n - 6 \\ n(n-1) &\leqslant 6n - 12 \\ n^2 - n &\leqslant 6n - 12 \\ n^2 &\leqslant 5n - 12. \end{aligned}$$

[69] Oops, the notation caught up with me. Here I use f for the number of faces, so I need a new letter to denote a face. I went with Greek.

But one can prove by induction that this inequality is false for all $n \geqslant 5$. Hence K_n is non-planar for all $n \geqslant 5$.[70]

You might think that the K in K_n stands for Komplett (complete in German), but apparently is stands for Kuratowski. You will prove on the homework that the complete bipartite graph $K_{3,3}$ is non-planar. It turns out that K_5 and $K_{3,3}$ are the "smallest non-planar graphs", in the following precise sense.

Kuratowski's Theorem (1930)

A graph is planar if and only if it does not contain a subgraph that is isomorphic to a subdivision of K_5 or $K_{3,3}$.

Never mind the details. I just included this for culture.

5.4 Circuits and Cycles

Section 5.2 discussed the existence and non-existence of paths in a graph. Now let us discuss the existence and non-existence of cycles.

Definition of Circuits and Cycles

Let $G = (V, E)$ be a graph.

- A *circuit* is a u, u-walk for some vertex $u \in V$. In other words, it is a walk that begins and ends at the same vertex.
- A *cycle* is a u, u-walk that contains no repetition, except for vertex u.

The *length* of a circuit or cycle is the number of times that it crosses an edge. Note that the basepoint u is arbitrary since a circuit/cycle can be based at any of its vertices.

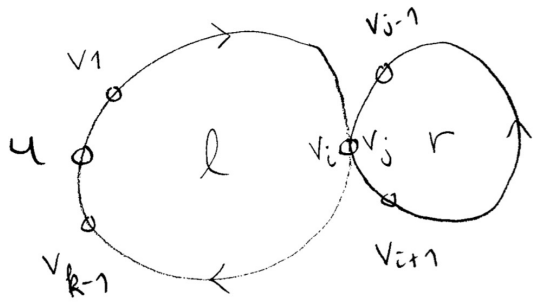
As with walks and paths, I claim that every u, u-circuit contains a u, u-cycle. Furthermore, I claim that any circuit of odd length contains a cycle of odd length, possibly with a different basepoint.

[70] Actually, we don't need to do this. If $n \geqslant 5$ then K_n contains K_5 as a subgraph. If we could draw K_n in the plane then we would obtain a planar drawing of K_5. Contradiction. Hence K_n cannot be drawn in the plane.

Proof by Induction on Length. Since we are working with simple graphs, the base case is a circuit of length 3, which is necessarily a cycle. Now consider a u, u-walk of length k:

$$u = v_0, v_1, \ldots, v_k = u.$$

If this is a cycle then we are done. Otherwise, there exists a repeated vertex $v_i = v_j$ with $1 \leqslant i < j \leqslant k-1$. Here is a picture (note that there may exist further repetitions that are not shown in the picture):



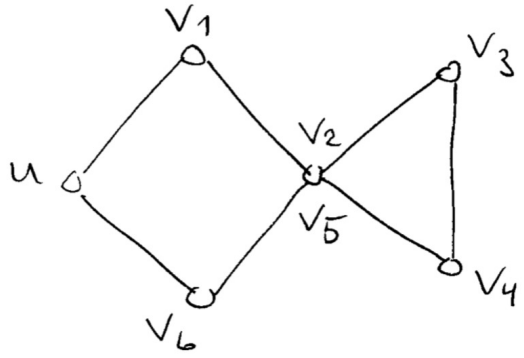
By induction on length, we may assume that the u, u-circuit labeled ℓ contains a u, u-cycle, which is also contained in the original circuit. This proves the first statement.

Now assume that k is an odd number and observe that

$$(\text{length of circuit } \ell) + (\text{length of circuit } r) = k.$$

It follows that exactly one of ℓ or r has odd length. Then by induction we may assume that this odd circuit contains an odd cycle. (Note that if r was the odd circuit then the odd cycle does not contain u.) □

For example, consider the following circuit $u = v_0, v_1, \ldots, v_6, v_7 = u$ of length 7:



This circuit contains the cycle $u = v_0, v_1, v_2, v_6, v_7 = u$ of length 4. Since 7 is odd we also know that there exists an odd cycle, in this case the 3-cycle v_2, v_3, v_4, v_2. However, this 3-cycle does not contain u. For our first application of the cycle concept we will discuss graphs that do not contain any **odd cycles**. The following theorem explains why these graphs are called "bipartite".

Characterization of Bipartite Graphs

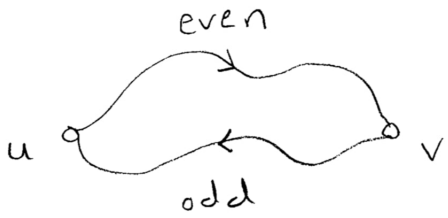
A graph $G = (V, E)$ is called *bipartite* if the vertices can be partitioned into two sets $V = A \cup B$ such that every edge has one endpoint in A and the other endpoint in B. Equivalently, I claim that

$$G \text{ is bipartite} \iff G \text{ has no odd cycles.}$$

For the purpose of the proof we may assume that the graph G is connected, since a disconnected graph is bipartite if and only if each of its components is bipartite. Similarly, a graph has no odd cycles if and only if each of its components has no odd cycles.

Proof. First suppose that G is bipartite with vertices $V = A \cup B$. Note that any cycle in the graph must bounce back and forth between the sets A and B, therefore the cycle must have even length.

Conversely, suppose that the graph G has no odd cycles. In this case we will prove that the graph is bipartite. For this purpose we select a basepoint $u \in V$ at random. I claim that for any given vertex $v \in V$, all u, v-walks have the same parity (i.e., they all have even length or they all have odd length). Indeed, suppose for contradiction that there exists an even u, v-walk and an odd u, v-walk:

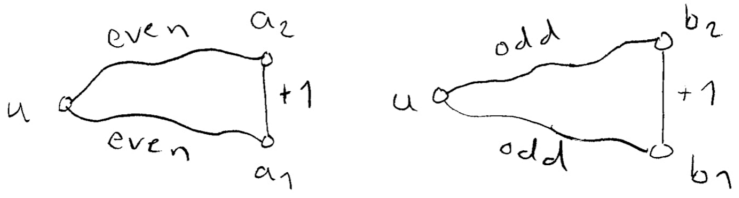


By reversing one of the walks and concatenating we obtain an odd circuit, which contains an odd cycle by the previous result. Contradiction. Thus we may partition the vertices into two disjoint sets:

$$A := \{v \in V : \text{every } u, v\text{-walk is even}\},$$

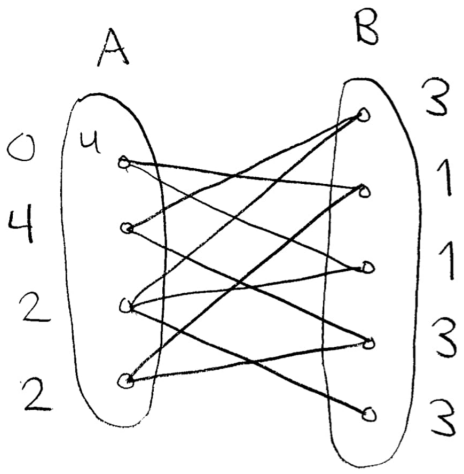
$$B := \{v \in V : \text{every } u, v\text{-walk is odd}\}.$$

(Note that $u \in A$ because 0 is an even number.) Finally, I claim that $V = A \cup B$ is a "bipartition" of the graph. Indeed, suppose for contradiction that there exists an edge $\{a_1, a_2\} \in E$ with $a_1, a_2 \in A$. Then by concatenating this edge with an (even) u, a_1-walk and an (even) u, a_2-walk we obtain an odd circuit, hence also an odd cycle. Contradiction. A similar argument shows that there can be no edge $\{b_1, b_2\} \in E$ with $b_1, b_2 \in B$:



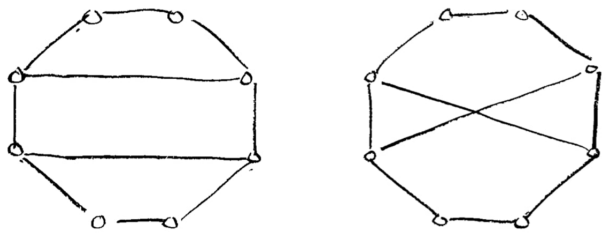
Hence we conclude that G is bipartite. □

For example, here is a connected bipartite graph:

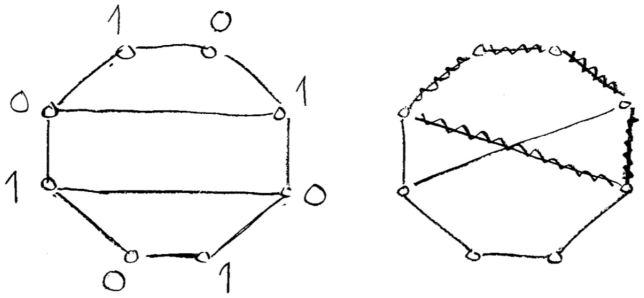


Each vertex is labeled with its shortest distance to the basepoint $u \in A$. Observe that there are no edges between the A vertices, nor between the B vertices. If we were to add all possible edges between A and B we would obtain the *complete bipartite graph* $K_{4,5}$.

As an application of the previous theorem we will prove that the following graphs are not isomorphic:



They certainly look different but how can we **prove** that they are different? The brute force attack would consider all of the $8! = 40320$ different labelings of the vertices and check that the edges never match up. Clearly we don't want to do this so we will use a trick. The following picture demonstrates that the graph on the left is bipartite, while the graph on the right has a 5-cycle:



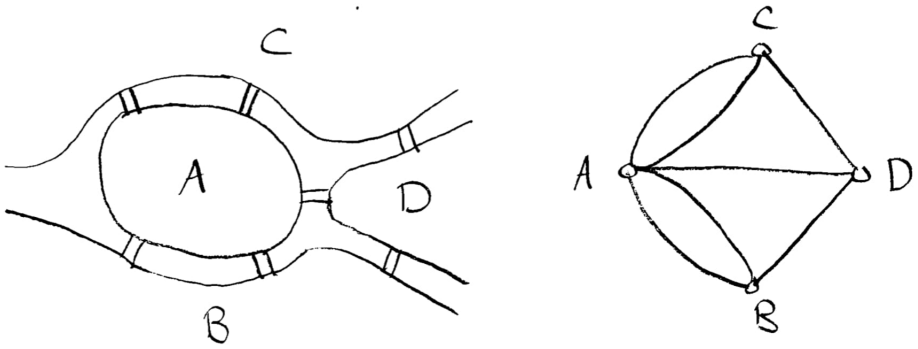
Since 5 is odd it follows from the previous theorem that the graph on the left does **not** contain a 5-cycle, hence the two graphs are not isomorphic.

Our next theorem is actually the oldest result in all of graph theory. It was published by Leonhard Euler in 1736.[71] As you will see, the subject has very humble beginnings.

In the Prussian city of Königsburg there was an island called Kneiphof in a river called Pregel.[72] In the year 1736 the surrounding 4 landmasses were connected by 7 bridges as in the following picture:

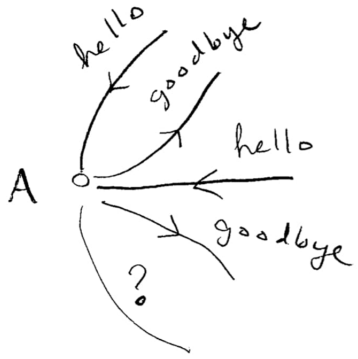
[71] *Solutio problematis ad geometriam situs pertinentis*, Proceedings of the Imperial Saint Petersburg Academy of Sciences, (1736).

[72] Königsburg was conquered by the Soviet Union in 1945 and renamed to Kaliningrad. Today it is still part of Russia.



The problem, which Euler says "is widely known", is to find a walk that crosses each bridge exactly once. For this purpose Euler realized that the geometry of the situation doesn't matter, only the connections. Thus we can view the landmasses as vertices and the bridges as edges. In the case of the Kneiphof we obtain the multigraph above. (For this problem, and this problem only, we will allow ourselves to discuss multigraphs.)

Euler observed that this problem is **impossible** for the following reason: Suppose for contradiction that your friend is able to take a walk, crossing each bridge exactly once. Now let them do the same walk again, while you sit at one of the vertices and watch. Make sure to choose a vertex that is not at the beginning or the end of the walk. Let's say you sit at vertex A which has 5 bridges (in graph theory language we say that $\deg(A) = 5$). Thus you will say "hello" to you friend twice and "goodbye" twice:



But now what happens? Since A is not at the start or the end of the walk, it is impossible for your friend to cross the 5th bridge at A without getting stuck. We conclude from this that every intermediate vertex of the walk (i.e., not the start or the end point) must have even degree. Since **Euler's graph has no vertex of even degee** this is impossible.

If Euler's only contribution was to solve this specific puzzle then it would be not be remembered today. However, Euler went on to state the following completely general theorem.

Euler's Theorem (1736)

Let $G = (V, E)$ be a connected graph, possibly with multiple edges. An *Euler walk* is a u, v-walk that crosses each edge exactly once.[73] We call this an *Euler circuit* when $u = v$. I claim that:

- There exists an Euler u, v-walk with $u \neq v$ if and only if u and v have odd degree and every other vertex has even degree.
- There exists an Euler circuit if and only if every vertex has even degree.

Proof. We have already observed that every vertex in an Euler walk that is not at the starting point or the endpoint must have even degree (every time you say "hello" to your friend you must also say "goodbye"). In an Euler circuit we can apply the same reasoning to show that **every** vertex in the graph must have even degree. This was the easy direction.

For the other direction, we need to prove that an Euler walk/circuit **always exists** under the right conditions.[74] First we will prove by induction on the number of edges that an Euler circuit always exists when the degree of each vertex is even. So let G be a connected graph in which every vertex has even degree and let $u \in V$ be a random vertex.[75] Apply the following algorithm to obtain a u, u-circuit:

procedure: to construct a u, u-circuit
$v_0 := u$
$v_1 :=$ any neighbor of v_0
while $v_k \neq u$ **do**
  $k := k + 1$
  $v_k :=$ any neighbor of v_{k-1}
  delete one copy of the edge $\{v_{k-1}, v_k\}$ from the graph
end do
return $(v_0, v_1, v_2, \ldots, v_k)$

Note that the algorithm never gets stuck since if $v_k \neq u$ then v_k always has a neighbor. Indeed, if at the kth step we have $v_k \neq u$ then we have deleted an odd number of vertices at v_k. Since $\deg(v_k)$ is even this means that v_k still has a neighbor. Furthermore, we know

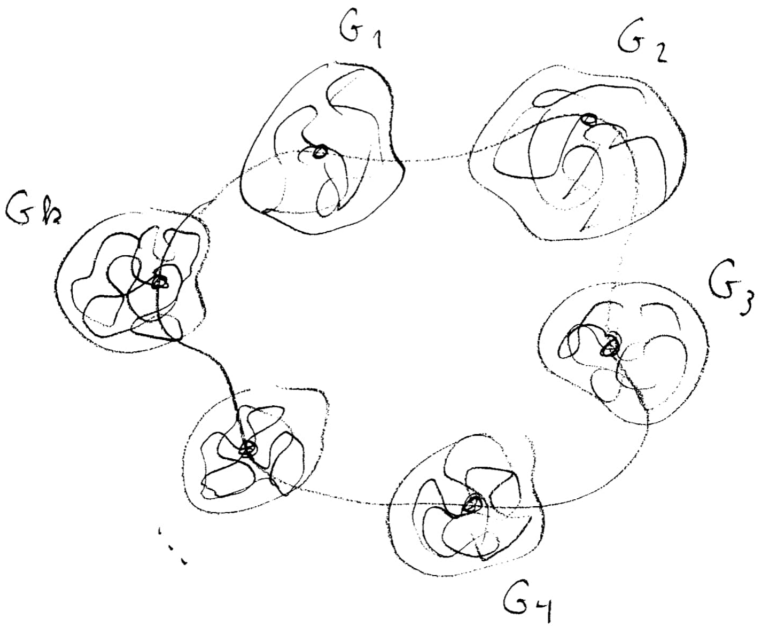
[73] In the case of multigraphs it is not enough to specify the sequence of vertices in a walk. We must also specify the sequence of edges. But never mind.

[74] Euler was a bit vague on this point, but that was the style in 1736.

[75] We also assume that the graph has more than one vertex and no loops.

that the algorithm must terminate because the graph has finitely many edges. Thus we have constructed a u, u-circuit (not necessarily Eulerian).

Now delete the edges of this circuit to obtain a graph G' with fewer edges, which might be disconnected. Let $G'_1, \ldots, G'_k$ be the components of G'. Since an even number of edges have been deleted at each vertex we observe that every vertex of G' is still even. Thus by induction we know that each component G'_i contains an Euler circuit. Finally we stitch each of these circuits into to the original u, u-circuit to obtain a big Euler circuit in G:



To complete the proof, let G be any connected graph with exactly two odd vertices $u, v \in V$. To prove that there exists an Euler u, v-walk, let us first add a new edge between u and v. Now every vertex has even degree and we conclude from the previous argument that there exists an Euler circuit. Finally, we delete the new edge from this circuit to obtain an Euler u, v-walk in the graph G. ▫

Here is a fun application.

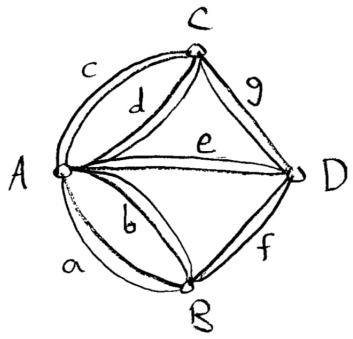
Every Graph has a Double Euler Circuit

In every graph G there exists a circuit that crosses each edge exactly **twice**.

Proof. Let G be a graph and let G' be the same graph with every edge doubled. It follows

that every vertex in G' has even degree, hence G' has an Euler circuit. This circuit in G' describes a circuit in G that crosses each edge exactly twice. □

For example, here is Euler's map of Königsburg with every bridge doubled:



One can verify that the following is a double Euler circuit:

$$AcCcAaBaAdCdAbBbAeDgCgDfBfDeA$$

5.5 Trees and Forests

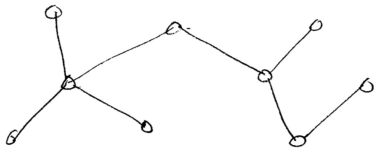
In this section we will consider graphs with very few edges. For example, suppose that G is a connected graph with n vertices. What is the fewest number of edges that G can have? (In other words: What is the fewest number of wires that you need to connect n computers?) Let e be the number of edges and let k be the number of connected components, and recall that we have proved the following inequality:

$$e \geqslant n - k.$$

In the case of a connected graph we have $k = 1$ and hence $e \geqslant n - 1$. Thus we conclude that

any connected graph on n vertices has at least $n - 1$ edges.

Graphs that attain this lower bound are called "trees". For example, here is a tree with 9 vertices (and hence 8 edges):



The interesting thing about trees is that one can define them in many different ways. For example, we will see below that the following definitions are all equivalent:

- A connected graph with the fewest possible number of edges.

- A connected graph with no cycles.
- A graph in which there exists a unique path between any two vertices.

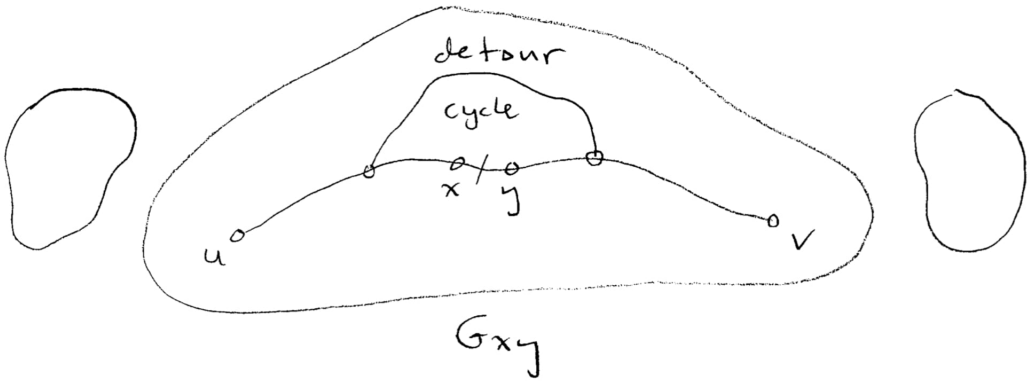
The following concept will help us in our study of trees.

Definition of a Bridge

Let $G = (V, E)$ be a graph. An edge $\{x, y\} \in E$ is called a *bridge* if by deleting it we increase the number of connected components.[76] I claim that deleting a bridge increases the number of components by exactly one. Furthermore, I claim that

$$\{x, y\} \text{ is a bridge} \iff \{x, y\} \text{ belongs to no cycle.}$$

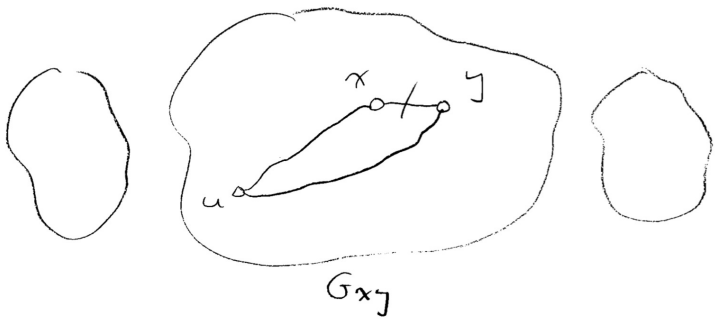
Proof. Let G_{xy} be the connected component containing the edge $\{x, y\}$ and let G' be the graph obtained from G by deleting the this edge. If $\{x, y\}$ belongs to a cycle in G then I claim that G_{xy} is still connected in G', hence the number of components remains the same. Indeed, consider any u, v-path in G_{xy} which contains the edge $\{x, y\}$. After deleting this edge we can still get from u to v by following a detour around the cycle:



Conversely, suppose that $\{x, y\}$ does not belong to a cycle in G. Then I claim that the component G_{xy} breaks into two components $G_{xy} = G_x \cup G_y$ in G' with $x \in G_x$ and $y \in G_y$. Indeed, suppose for contradiction that G_{xy} is still connected in G'. Then by definition there exists an x, y-path in G'. But this path together with the edge $\{x, y\}$ would create a cycle in G. Contradiction. We conclude that the number of components goes up, hence $\{x, y\}$ is a bridge. Finally, we will show that the number of components goes up **by exactly one**. For

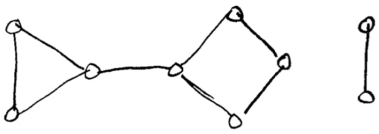
[76] Do not confuse the formal concept of a "bridge" with the bridges of Königsberg, which are merely edges.

this we will prove that every vertex $u \in G_{xy}$ is still connected to one of x or y in G'. Indeed, since $u, y \in G_{xy}$ we know that there exists a u, y-path in G. If this path does not use the edge $\{x, y\}$ then the u, y-path still exists in G'. On the other hand, if the u, y-path in G uses the edge $\{x, y\}$ then there still exists a u, x-path in G':



□

For example, the following graph has exactly two bridges. Can you find them?



Before officially defining a "tree", it is convenient to define a "forest".

Characterization of Forests

Let G be a graph with n vertices, e edges and k connected components. Recall from the Connected Components Theorem that we must have $e \geqslant n - k$. I claim that the number of edges is minimized precisely when the graph has no cycles:

$$e = n - k \quad \Longleftrightarrow \quad G \text{ has no cycles.}$$

In either case we say that G is a *forest*.

Proof. First suppose that $e = n - k$ and assume for contradiction that G has a cycle. Let G' be the graph obtained by deleting a random edge from this cycle and let n', e', k' be the

parameters of this new graph, so that $n' = n$ and $e' = e - 1$. Since the edge we deleted was in a cycle, it was not a bridge. Thus we have $k = k'$ and hence

$$e' < e = n - k = n' - k'$$

which contradicts the fact $e' \geqslant n' - k'$.

Conversely, suppose that G has no cycles. We will prove by induction on the number of edges that $e = n - k$. Indeed, if $e = 0$ then we must have $n = k$ and hence $n - k = 0 = e$. Now suppose that $e \geqslant 1$ and delete a random edge to obtain a graph G' with parameters n', e', k' so that $n' = n$ and $e' = e - 1$. Since G has no cycles we know that the deleted edge was a bridge and hence $k' = k - 1$. Furthermore, since G has no cycles we see that G' also has no cycles. Thus we conclude by induction that $e' = n' - k'$ and hence

$$e = e' + 1 = (n' - k') + 1 = (n - (k + 1)) + 1 = n - k$$

as desired. □

Finally I will give the official definition of a tree. As I mentioned above, there are at least three different ways to state the definition. A computer science professor told me that he really wants you to learn this theorem because it is important for the analysis of algorithms.

Theorem/Definition of Trees

I claim that the following definitions are equivalent:

(T1) A connected forest.

(T2) A connected graph with the least possible number of edges.

(T3) A connected graph with no cycles.

(T4) A graph in which there exists a unique path between any two vertices.

Any graph satisfying one (and hence all) of these definitions is called a *tree*. Observe that every connected component of a forest is a tree, hence the name.

The proof will be terse because we already did all of the work.

Proof. (T1)⇒(T2): A connected forest has $k = 1$ and $e = n - k = n - 1$. This the minimum possible number of edges. (T2)⇒(T3): Consider a connected graph ($k = 1$) with $e = n-1$. From the Characterization of Forests this graph has no cycles. (T3)⇒(T4): Consider a connected graph with no cycles. By connectivity there exists a u, v-path for all $u, v \in V$. If there existed two such paths then we would obtain a cycle. Contradiction. (T4)⇒(T1):

Consider a graph in which there exists a unique path between any two vertices. Such a graph is connected. Furthermore, such a graph has no cycles, since any two vertices in a cycle are connected by two paths. Hence the graph is a connected forest. ▫

I will end this section with an important application. A graph can represent a network of roads between cities. In this case, each edge/road has an associated distance/cost. It is natural to ask what is the cheapest way to connect all of the cities. In this case we are looking for a "minimum cost spanning tree", i.e., a tree whose edges connect all of the vertices, and whose edges have the minimum possible cost. It turns out that the most obvious possible algorithm gives a valid solution, and is reasonably efficient.

Kruskal's Algorithm (1956)

Let $G = (V, E)$ be a connected graph and suppose that each edge $e \in E$ has an associated number $w(e) \in \mathbb{R}$ called its *weight*. Our goal is to find a *mimimum weight spanning tree*, i.e., a connected subgraph with the minimum number of possible edges, in which the sum of the edge weights is minimized. I claim that the following algorithm works:

procedure: to find a minimum weight spanning tree for $G = (V, E_G)$
$E_T := \{\}$
sort edges $E_G = \{e_1, \ldots, e_m\}$ so that $w(e_i) \leqslant w(e_{i+1})$ for all i
for i **from** 1 **to** m **do**
  if $E_T \cup \{e_i\}$ has no cycle **then**
    $E_T := E_T \cup \{e_i\}$
  end if
end do
return $T = (V, E_T)$

Proof. Let $T = (V, E_T)$ be the graph output by the algorithm and note that the subset $E_T \subseteq E_G$ has the property that adding one further edge creates a cycle. First we will show that T is a spanning tree. To do this, suppose for contradiction that there exist two vertices $u, v \in V$ such that there is no u, v-path in T. Since G is connected there exists is a u, v-path in G, which must use some edge that is not in E_T. By adding this edge to E_T we obtain a cycle in T, which implies that there was already a u, v-path in T. Contradiction. Thus T is a connected graph with vertex set V and no cycles. In other words, it is a spanning tree.

Next we will prove by induction that the spanning tree T has minimum weight. To do this, let T_1 be **any** spanning tree of minimum weight. If $T_1 = T$ then we are done. Otherwise, we will show how to construct another spanning tree T_2 such that

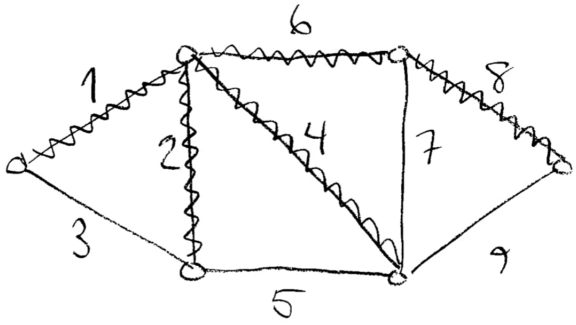
- T_2 shares one more edge in common with T than T_1 does,

- weight(T_2) = weight(T_1), hence T_2 also has minimum weight.

We can repeat this construction to obtain a sequence of minimum weight spanning trees $T_1, T_2, \ldots$, each of which has one more edge in common with T. After a finite number of steps we will obtain $T_k = T$ and we will conclude that T itself has minimum weight.

Here is the construction: Let $T_i = (V, E_{T_i})$ be any minimum weight spanning tree. If $T_i = T$ then we are done, so let $e' \in E_{T_i}$ be the first edge of T_i that was not chosen by Kruskal's algorithm. By adding e' to E_T we obtain a cycle, which since T is acyclic necessarily contains some edge $e \in E_T$ that is not in E_{T_i}. Since $T - e + e'$ still has $\#V - 1$ edges, it is still a tree and hence has no cycles. This means that Kruskal's algorithm could have chosen the edge e' instead of e. The fact that it didn't implies that $w(e) \leqslant w(e')$. We conclude that $T_{i+1} := T_i - e' + e$ is a spanning tree that has one more edge in common with T and also satisfies weight(T_{i+1}) $\leqslant$ weight(T_i). Since T_i had minimum weight this implies that weight(T_{i+1}) = weight(T_i) and hence T_{i+1} also has minimum weight. □

Kruskal's algorithm is an example of a *greedy algorithm*, i.e., an algorithm that makes the most obvious choice at each step. Sometimes a greedy algorithm can get stuck with a non-optimal solution. We are lucky when a greedy algorithm (such as Krukal's) leads to an optimal solution. Here's an example:



Note that the edges of weight $1, 2, 4, 6, 8$ were chosen in increasing order. It follows that every other spanning tree for this graph has weight greater than or equal to $1 + 2 + 4 + 6 + 8 = 21$.

Example.

5.6 Counting Trees and Forests

We have seen that Kruskal's algorithm produces a minimum weight spanning tree for a given weighted graph. It is natural to wonder, therefore, how many different spanning trees a graph can have.

I should investigate the history of the Matrix Tree Theorem. How is it implicit in Kirchoff's work on electrical circuits (1847).

Cayley's Tree Formula (1881). Prüfer's proof (1918).

The number of spanning trees of $K_{m,n}$ is $m^{n-1}n^{m-1}$.[77]

Other kinds of graphs are too hard to count.

5.7 Worked Exercises

5.1. Check that the relation $\sum_{u \in V_G} \deg(u) = 2 \cdot \#E_G$ holds for each of the following:

(a) Cycles C_n

(b) Paths P_n

(c) Complete graphs K_n

(d) Complete bipartite graphs $K_{m,n}$

(a) The graph C_n has n vertices, each with degree 2. Hence the degree sum is

$$\sum_{u \in V} \deg(u) = \underbrace{2 + 2 + \cdots + 2}_{n \text{ times}} = 2n.$$

On the other hand, the number of edges is $\#E = n$, hence $2 \cdot \#E = 2n$.

(b) The path P_n has two vertices of degree 1 and $n-2$ vertices of degree 2, hence

$$\sum_{u \in V} \deg(u) = 1 + \underbrace{2 + \cdots + 2}_{n-2 \text{ times}} + 1 = 1 + 2(n-2) + 1 = 2(n-1).$$

On the other hand, the path P_n has $\#E = n-1$ (indeed, it is a tree) and hence $2 \cdot \#E = 2(n-1)$.

(c) The complete graph K_n has n vertices, each with degree $n-1$, hence

$$\sum_{u \in V} \deg(u) = \underbrace{(n-1) + \cdots + (n-1)}_{n \text{ times}} = n(n-1).$$

On the other hand, the number of edges is $\#E = \binom{n}{2}$ because there is an edge between each pair of vertices. This agrees with the degree sum because

$$2 \cdot \#E = 2\binom{n}{2} = 2 \cdot \frac{n(n-1)(n-2)(n-3)\cdots 3 \cdot 2 \cdot 1}{2 \cdot 1 \cdot (n-2)(n-3)\cdots 3 \cdot 2 \cdot 1} = 2 \cdot \frac{n(n-1)}{2} = n(n-1).$$

[77] https://math.stackexchange.com/questions/3157546/prove-that-the-complete-bipartite-graph-k-3-s-has-s23s-1-sp

(d) The complete bipartite graph $K_{m,n}$ has vertices $V = A \cup B$ with $\#A = m$ and $\#B = n$. Each vertex in A has degree n (because it connects to each vertex in B) and each vertex in B has degree m (because it connects to each vertex in A). Hence the degree sum is

$$\sum_{u \in V} \deg(u) = \sum_{a \in A} \deg(a) + \sum_{b \in B} \deg(b) = mn + nm = 2mn.$$

On the other hand, the number of edges in $K_{m,n}$ is mn because each edge has exactly one vertex in A, hence $\#E = \sum_{a \in A} \deg(a) = mn$. This agrees with the degree sum. Remark: The identity

$$\sum_{a \in A} \deg(a) = mn = \sum_{b \in B} \deg(b)$$

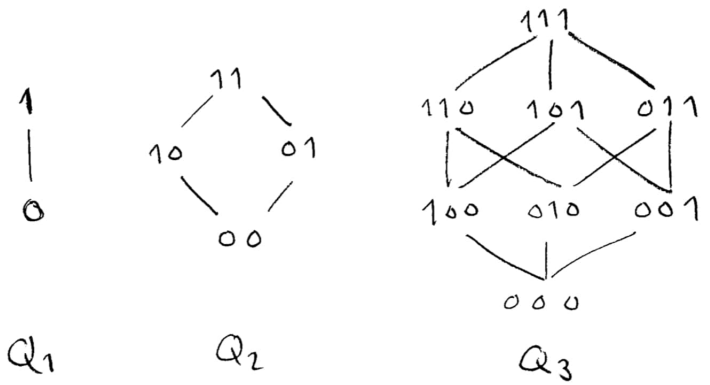
will show up again in Problem 5.6.

5.2. The *hypercube graph* Q_n has 2^n vertices corresponding to the binary strings of length n and edges corresponding to "flipping one bit".

(a) Draw the graphs Q_1, Q_2, Q_3.

(b) Compute the number of edges in Q_n. [Hint: What are the vertex degrees?]

(a) Here they are:



(b) We know from the previous chapter that there are 2^n binary strings of length n. Hence the graph Q_n has 2^n vertices. Furthermore, since each edge corresponds to "flipping a bit", and since each vertex has n possible bits to flip, we see that each vertex in Q_n has degree n. Finally, the Handshaking Lemma tells us that

$$2 \cdot \#E = \sum_{u \in V} \deg(u)$$

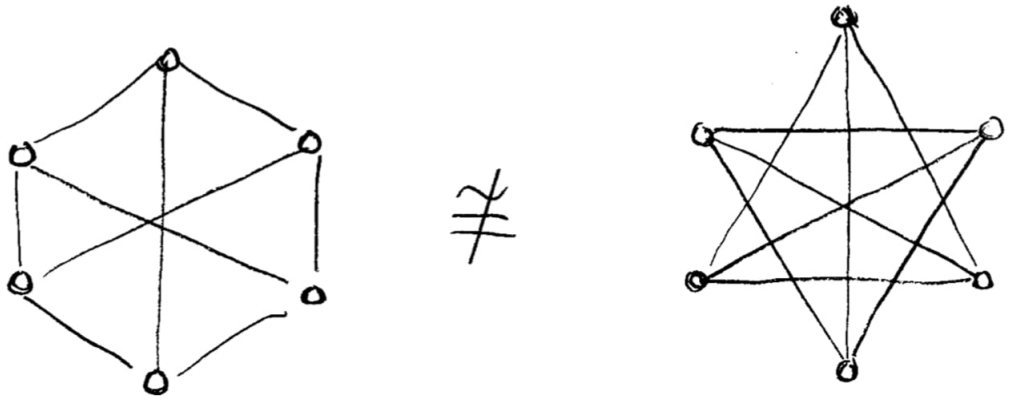
$$2 \cdot \#E = \underbrace{n + n + \cdots + n}_{2^n \text{ times}}$$
$$2 \cdot \#E = n2^n$$
$$\#E = n2^{n-1}.$$

Where would we be without the Handshaking Lemma??

5.3. Explain why every (simple) graph has two vertices of the same degree. [Hint: Suppose that the graph has n vertices. Show that the degrees 0 and $n-1$ cannot both occur. So how many possible degree values are there?]

Proof. Each vertex of a simple graph on n vertices must have its degree in the set $\{0, 1, \ldots, n\}$. Furthermore, it is impossible to have two vertices with degrees 0 and $n-1$. Indeed, if there exists a vertex of degree $n-1$ then it shares an edge with every other vertex, but a vertex of degree 0 shares an edge with no one. Therefore there are at most $n-1$ different possible degrees (pigeonholes). Since there are n total vertices (pigeons), two vertices must share a degree (two pigeons must share a hole).[78] □

5.4. Give two different proofs that the following graphs are not isomorphic:

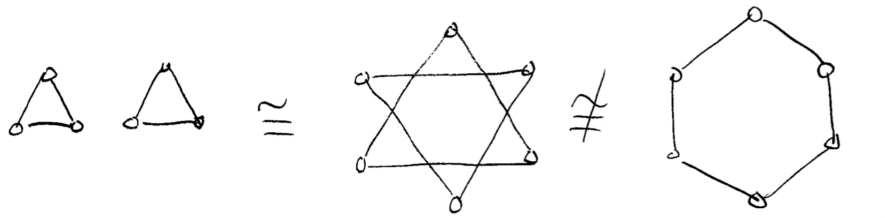


(a) Show that the complements are not isomorphic.

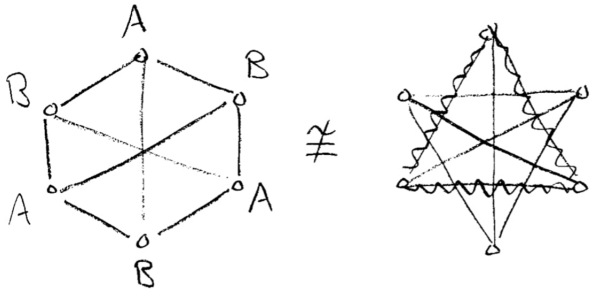
(b) Show that the left graph is bipartite, while the right graph is not.

(a) Here are the complements. Note that the left graph has two connected components, while the right graph has one connected component. Since the complements are not isomorphic the original graphs are also not isomorphic.

[78] Yes, this type of argument is called the *pigeonhole principle*. I don't know where the name comes from.



The following picture demonstrates that the right graph has a 3-cycle, while the left graph is bipartite (hence does not have a 3-cycle):



5.5. Let G be a (simple) graph with n vertices.

(a) If $2 \leqslant k \leqslant n$ show that $\binom{n-k+1}{2} \leqslant \binom{n-1}{2}$.

(b) If G has more than $\binom{n-1}{2}$ edges, prove that G is connected. [Hint: Let k be the number of connected components of G. There is a relevant theorem in the notes.]

(c) Draw a graph with 6 vertices and $\binom{5}{2}$ edges that is **not** connected.

(a) More generally, consider any integers $2 \leqslant m \leqslant n$. Then we have

$$m(m-1) \leqslant n(m-1) \leqslant n(n-1)$$

and hence

$$\binom{m}{2} = \frac{m(m-1)}{2} \leqslant \frac{n(n-1)}{2} = \binom{n}{2}.$$

(b) **Proof.** We proved in class that any (simple) graph with n vertices, e edges and k connected components satisfies

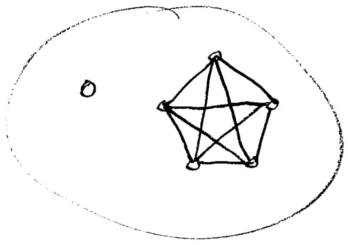
$$e \leqslant \binom{n-k+1}{2}.$$

If $k \geqslant 2$ then it follows from part (a) that

$$e \leqslant \binom{n-k+1}{2} \leqslant \binom{n-1}{2}.$$

Therefore if $e > \binom{n-1}{2}$ then we must have $k < 2$, hence G is connected. □

(c) Here is disconnected graph on 6 vertices with the **maximum** number of edges:



5.6. Let $G = (V, E)$ be a bipartite graph with partition $V = A \cup B$. In other words, assume that every edge of the graph has the form $\{a, b\}$ for some $a \in A$ and $b \in B$.

(a) Let $\deg(A), \deg(B)$ be the average degree of a vertex in A, B, respectively. Prove that

$$\#A \cdot \deg(A) = \#B \cdot \deg(B).$$

(b) A certain statistical survey[79] found that men in the United States report 74% more opposite sex partners than women. Explain why this statistic cannot possibly be accurate. [Hint: Let A and B be the sets of men and women.]

(a) We will count the edges in two ways. On the one hand, every edge has exactly one endpoint in A, hence

$$\#E = \sum_{a \in A} \#(\text{edges containing } a) = \sum_{a \in A} \deg(a).$$

On the other hand, every edge has exactly one endpoint in B, hence

$$\#E = \sum_{b \in B} \#(\text{edges containing } b) = \sum_{b \in B} \deg(b).$$

It follows that the average degrees $\deg(A)$ and $\deg(B)$ satisfy

$$\sum_{a \in A} \deg(a) = \sum_{b \in B} \deg(b)$$

[79] *The Social Organization of Sexuality* (1994) by Edward O. Laumann et al. The authors themselves acknowledge (pg. 185) that this result cannot be accurate.

$$\#A\left(\frac{\sum_{a\in A}\deg(a)}{\#A}\right) = \#B\left(\frac{\sum_{b\in B}\deg(b)}{\#B}\right)$$
$$\#A\cdot\deg(A) = \#B\cdot\deg(B).$$

(b) Let A, B be the sets of men and women in the United States. Let $G = (A \cup B, E)$ be the graph whose edges are "opposite sex partnerships". Since this graph is bipartite we know from part (a) that

$$\frac{\deg(A)}{\deg(B)} = \frac{\#B}{\#A} = \frac{\#\text{ women}}{\#\text{ men}} \approx 1.$$

However, the survey found that

$$\frac{\deg(A)}{\deg(B)} = 1.74.$$

There is no way this can be accurate.

5.7. (This was not assigned.)

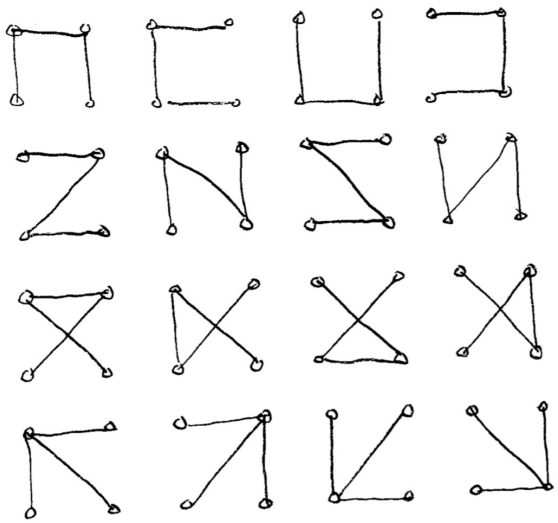
(a) We proved above that the number of trees on the vertex set $\{1,2,3,4\}$ is $4^{4-2} = 16$. Draw them all.

(b) We also proved that the number of trees on $\{1,2,3,4,5\}$ with degrees $1,1,1,2,3$ equals

$$\binom{5-2}{1-1,1-1,1-1,2-1,3-1} = \binom{3}{0,0,0,1,2} = \frac{3!}{1!2!} = 3.$$

Draw them.

(a) Here are the 16 trees on $\{1,2,3,4\}$:



Observe that the number of trees with degrees $1, 1, 2, 2$ (in some order) is

$$\binom{2}{0,0,1,1} + \binom{2}{0,1,0,1} + \cdots + \binom{2}{1,1,0,0} = 6 \cdot 2 = 12$$

and the number of trees with degrees $1, 1, 1, 3$ (in some order) is

$$\binom{2}{0,0,0,2} + \binom{2}{0,0,2,0} + \binom{2}{0,0,2,0} + \binom{2}{2,0,0,0} = 4 \cdot 1 = 4.$$

(b) Here are the 3 trees on $\{1, 2, 3, 4, 5\}$ with degrees $1, 1, 1, 2, 3$: