Discrete Mathematics

Lecture Notes, Yale University, Spring 1999

L. Lovász and K. Vesztergombi

Parts of these lecture notes are based on
L. Lovász – J. Pelikán – K. Vesztergombi: Kombinatorika
(Tankönyvkiadó, Budapest, 1972);
Chapter 13 is based on a section in
L. Lovász – M.D. Plummer: Matching theory
(Elsevier, Amsterdam, 1979)

Contents

1 Introduction

For most students, the first and often only area of mathematics in college is calculus. And it is true that calculus is the single most important field of mathematics, whose emergence in the 17th century signalled the birth of modern mathematics and was the key to the successful applications of mathematics in the sciences.

But calculus (or analysis) is also very technical. It takes a lot of work even to introduce its fundamental notions like continuity or derivatives (after all, it took 2 centuries just to define these notions properly). To get a feeling for the power of its methods, say by describing one of its important applications in detail, takes years of study.

If you want to become a mathematician, computer scientist, or engineer, this investment is necessary. But if your goal is to develop a feeling for what mathematics is all about, where is it that mathematical methods can be helpful, and what kind of questions do mathematicians work on, you may want to look for the answer in some other fields of mathematics.

There are many success stories of applied mathematics outside calculus. A recent hot topic is mathematical cryptography, which is based on number theory (the study of positive integers $1, 2, 3, \ldots$), and is widely applied, among others, in computer security and electronic banking. Other important areas in applied mathematics include linear programming, coding theory, theory of computing. The mathematics in these applications is collectively called *discrete mathematics*. ("Discrete" here is used as the opposite of "continuous"; it is also often used in the more restrictive sense of "finite".)

The aim of this book is not to cover "discrete mathematics" in depth (it should be clear from the description above that such a task would be ill-defined and impossible anyway). Rather, we discuss a number of selected results and methods, mostly from the areas of combinatorics, graph theory, and combinatorial geometry, with a little elementary number theory.

At the same time, it is important to realize that mathematics cannot be done without *proofs*. Merely stating the facts, without saying something about why these facts are valid, would be terribly far from the spirit of mathematics and would make it impossible to give any idea about how it works. Thus, wherever possible, we'll give the proofs of the theorems we state. Sometimes this is not possible; quite simple, elementary facts can be extremely difficult to prove, and some such proofs may take advanced courses to go through. In these cases, we'll state at least that the proof is highly technical and goes beyond the scope of this book.

Another important ingredient of mathematics is *problem solving*. You won't be able to learn any mathematics without dirtying your hands and trying out the ideas you learn about in the solution of problems. To some, this may sound frightening, but in fact most people pursue this type of activity almost every day: everybody who plays a game of chess, or solves a puzzle, is solving discrete mathematical problems. The reader is strongly advised to answer the questions posed in the text and to go through the problems at the end of each chapter of this book. Treat it as puzzle solving, and if you find some idea that you come up with in the solution to play some role later, be satisfied that you are beginning to get the essence of how mathematics develops.

We hope that we can illustrate that mathematics is a building, where results are built on earlier results, often going back to the great Greek mathematicians; that mathematics is alive, with more new ideas and more pressing unsolved problems than ever; and that mathematics is an art, where the beauty of ideas and methods is as important as their difficulty or applicability.

2 Let us count!

2.1 A party

Alice invites six guests to her birthday party: Bob, Carl, Diane, Eve, Frank and George. When they arrive, they shake hands with each other (strange european costum). This group is strange anyway, because one of them asks: "How many handshakes does this mean?"

"I shook 6 hands altogether" says Bob, "and I guess, so did everybody else."

"Since there are seven of us, this should mean $7 \cdot 6 = 42$ handshakes" ventures Carl.

"This seems too many" says Diane. "The same logic gives 2 handshakes if two persons meet, which is clearly wrong."

"This is exactly the point: every handshake was counted twice. We have to divide 42 by 2, to get the right number: 21." settles Eve the issue.

When they go to the table, Alice suggests:

"Let's change the seating every half an hour, until we get every seeting."

"But you stay at the head of the table" says George, "since you have your birthday."

How long is this party going to last? How many different seatings are there (with Alice's place fixed)?

Let us fill the seats one by one, starting with the chair on Alice's right. We can put here any of the 6 guests. Now look at the second chair. If Bob sits on the first chair, we can put here any of the remaining 5 guests; if Carl sits there, we again have 5 choices, etc. So the number of ways to fill the first two chairs is $5 + 5 + 5 + 5 + 5 + 5 = 6 \cdot 5 = 30$. Similarly, no matter how we fill the first two chairs, we have 4 choices for the third chair, which gives $6 \cdot 5 \cdot 4$ ways to fill the first three chairs. Going on similarly, we find that the number of ways to seat the guests is $6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 720$.

If they change seats every half an hour, it takes 360 hours, that is, 15 days to go through all seating orders. Quite a party, at least as the duration goes!

2.1 How many ways can these people be seated at the table, if Alice too can sit anywhere?

After the cake, the crowd wants to dance (boys with girls, remember, this is a conservative european party). How many possible pairs can be formed?

OK, this is easy: there are 3 girls, and each can choose one of 4 guys, this makes $3 \cdot 4 = 12$ possible pairs.

After about ten days, they really need some new ideas to keep the party going. Frank has one:

"Let's pool our resources and win a lot on the lottery! All we have to do is to buy enough tickets so that no matter what they draw, we should have a ticket with the right numbers. How many tickets do we need for this?"

(In the lottery they are talking about, 5 numbers are selected from 90.)

"This is like the seating" says George, "Suppose we fill out the tickets so that Alice marks a number, then she passes the ticket to Bob, who marks a number and passes it to Carl, ... Alice has 90 choices, no matter what she chooses, Bob has 89 choices, so there are $90 \cdot 89$ choices for the first two numbers, and going on similarly, we get $90 \cdot 89 \cdot 88 \cdot 87 \cdot 86$ possible choices for the five numbers."

"Actually, I think this is more like the handshake question" says Alice. "If we fill out the tickets the way you suggested, we get the same ticket more then once. For example, there will be a ticket where I mark 7 and Bob marks 23, and another one where I mark 23 and Bob marks 7."

Carl jumped up:

"Well, let's imagine a ticket, say, with numbers $7, 23, 31, 34$ and 55. How many ways do we get it? Alice could have marked any of them; no matter which one it was that she marked, Bob could have marked any of the remaining four. Now this is really like the seating problem. We get every ticket $5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$ times."

"So" concludes Diane, "if we fill out the tickets the way George proposed, then among the $90 \cdot 89 \cdot 88 \cdot 87 \cdot 86$ tickets we get, every 5-tuple occurs not only once, but $5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$ times. So the number of *different* tickets is only

$$\frac{90 \cdot 89 \cdot 88 \cdot 87 \cdot 86}{5 \cdot 4 \cdot 3 \cdot 2 \cdot 1}.$$

We only need to buy this number of tickets."

Somebody with a good pocket calculator computed this value in a glance; it was 43,949,268. So they had to decide (remember, this happens in a poor european country) that they don't have enough money to buy so many tickets. (Besides, they would win much less. And to fill out so many tickets would spoil the party...)

So they decide to play cards instead. Alice, Bob, Carl and Diane play bridge. Looking at his cards, Carl says: "I think I had the same hand last time."

"This is very unlikely" says Diane.

How unlikely is it? In other words, how many different hands can you have in bridge? (The deck has 52 cards, each player gets 13.) I hope you have noticed it: this is essentially the same question as the lottery. Imagine that Carl picks up his cards one by one. The first card can be any one of the 52 cards; whatever he picked up first, there are 51 possibilities for the second card, so there are $52 \cdot 51$ possibilities for the first two cards. Arguing similarly, we see that there are $52 \cdot 51 \cdot 50 \cdot \ldots \cdot 40$ possibilities for the 13 cards.

But now every hand was counted many times. In fact, if Eve comes to quibbiz and looks into Carl's cards after he arranged them, and tries to guess (I don't now why) the order in which he picked them up, she could think: "He could have picked up any of the 13 cards first; he could have picked up any of the remaining 12 cards second; any of the remaining 11 cards third;... Aha, this is again like the seating: there are $13 \cdot 12 \cdot \ldots \cdot 2 \cdot 1$ orders in which he could have picked up his cards."

But this means that the number of *different* hands in bridge is

$$\frac{52 \cdot 51 \cdot 50 \cdot \ldots \cdot 40}{13 \cdot 12 \cdot \ldots \cdot 2 \cdot 1} = 635{,}013{,}559{,}600.$$

So the chance that Carl had the same hand twice in a row is one in 635,013,559,600, very small indeed.

Finally, the six guests decide to play chess. Alice, who just wants to watch them, sets up 3 boards.

"How many ways can you guys be matched with each other?" she wonders. "This is clearly the same problem as seating you on six chairs; it does not matter whether the chairs are around the dinner table of at the three boards. So the answer is 720 as before."

"I think you should not count it as a different matching if two people at the same board switch places" says Bob, "and it should not matter which pair sits at which table."

"Yes, I think we have to agree on what the question really means" adds Carl. "If we include in it who plays white on each board, then if a pair switches places we do get a different matching. But Bob is right that it does not matter which pair uses which board."

"What do you mean it does not matter? You sit at the first table, which is closest to the peanuts, and I sit at the last, which is farthest" says Diane.

"Let's just stick to Bob's version of the question" suggests Eve. "It is not hard, actually. It is like with handshakes: Alice's figure of 720 counts every matching several times. We could rearrange the tables in 6 different ways, without changing the matching."

"And each pair may or may not switch sides" adds Frank. "This means $2 \cdot 2 \cdot 2 = 8$ ways to rearrange people without changing the matching. So in fact there are $6 \cdot 8 = 48$ ways to sit which all mean the same matching. The 720 seatings come in groups of 48, and so the number of matchings is $720/48 = 15$."

"I think there is another way to get this" says Alice after a little time. "Bob is youngest, so let him choose a partner first. He can choose his partner in 5 ways. Whoever is youngest among the rest, can choose his or her partner in 3 ways, and this settles the matching. So the number of matchings is $5 \cdot 3 = 15$."

"Well, it is nice to see that we arrived at the same figure by two really different arguments. At the least, it is reassuring" says Bob, and on this happy note we leave the party.

2.2 What is the number of "matchings" in Carl's sense (when it matters who sits on which side of the board, but the boards are all alike), and in Diane's sense (when it is the other way around)?

2.2 Sets and the like

We want to formalize assertions like "the problem of counting the number of hands in bridge is essentially the same as the problem of counting tickets in the lottery". The usual tool in mathematics to do so is the notion of a *set*. Any collection of things, called *elements*, is a set. The deck of cards is a set, whose elements are the cards. The participants of the party form a set, whose elements are Alice, Bob, Carl, Diane, Eve, Frank and George (let us denote this set by P). Every lottery ticket contains a set of 5 numbers.

For mathematics, various sets of numbers are important: the set of real numbers, denoted by $\mathbb{R}$; the set of rational numbers, denoted by $\mathbb{Q}$; the set of integers, denote by $\mathbb{Z}$; the set of non-negative integers, denoted by $\mathbb{Z}_+$; the set of positive integers, denoted by $\mathbb{N}$. The *empty set*, the set with no elements is another important (although not very interesting) set; it is denoted by $\emptyset$.

If A is a set and b is an element of A, we write $b \in A$. The number of elements of a set A (also called the *cardinality* of A) is denoted by $|A|$. Thus $|P| = 7$; $|\emptyset| = 0$; and $|\mathbb{Z}| = \infty$ (infinity).[1]

We may specify a set by listing its elements between braces; so

$$P = \{\text{Alice, Bob, Carl, Diane, Eve, Frank, George}\}$$

is the set of participants of Alice's birthday party, or

$$\{12, 23, 27, 33, 67\}$$

[1]In mathematics, one can distinguish various levels of "infinity"; for example, one can distinguish between the cardinalities of $\mathbb{Z}$ and $\mathbb{R}$. This is the subject matter of *set theory* and does not concern us here.

is the set of numbers on my uncle's lottery ticket. Sometimes we replace the list by a verbal description, like

$$\{\text{Alice and her guests}\}.$$

Often we specify a set by a property that singles out the elements from a large universe like real numbers. We then write this property inside the braces, but after a colon. Thus

$$\{x \in \mathbb{Z} : \ x \geq 0\}$$

is the set of non-negative integers (which we have called $\mathbb{Z}_+$ before), and

$$\{x \in P : \ x \text{ is a girl}\} = \{\text{Alice, Diane, Eve}\}$$

(we denote this set by G). Let me also tell you that

$$D = \{x \in P : \ x \text{ is over } 21\} = \{\text{Alice, Carl, Frank}\}$$

(we denote this set by D).

A set A is called a *subset* of a set B, if every element of A is also an element of B. In other words, A consists of certain elements of B. We allow that A consists of all elements of B (in which case $A = B$), or none of them (in which case $A = \emptyset$). So the empty set is a subset of every set. The relation that A is a subset of B is denoted by

$$A \subseteq B.$$

For example, among the various sets of people considered above, $G \subseteq P$ and $D \subseteq P$. Among the sets of numbers, we have a long chain:

$$\emptyset \subseteq \mathbb{N} \subseteq \mathbb{Z}_+ \subseteq \mathbb{Z} \subseteq \mathbb{Q} \subseteq \mathbb{R}$$

The *intersection* of two sets is the set consisting of those elements that elements of both sets. The intersection of two sets A and B is denoted by $A \cap B$. For example, we have $G \cap D = \{Alice\}$. Two sets whose intersection is the empty set (in other words, have no element in common) are called *disjoint.*

2.3 Name sets whose elements are (a) buildings, (b) people, (c) students, (d) trees, (e) numbers, (f) points.

2.4 What are the elements of the following sets: (a) army, (b) mankind, (c) library, (d) the animal kingdom?

2.5 Name sets having cardinality (a) 52, (b) 13, (c) 32, (d) 100, (e) 90, (f) 2,000,000.

2.6 What are the elements of the following (admittedly peculiar) set: $\{\text{Alice}, \{1\}\}$?

2.7 We have not written up all subset relations between various sets of numbers; for example, $\mathbb{Z} \subseteq \mathbb{R}$ is also true. How many such relations can you find between the sets $\emptyset, \mathbb{N}, \mathbb{Z}_+, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$?

2.8 Is an "element of a set" a special case of a "subset of a set"?

2.9 List all subsets of $\{0, 1, 3\}$. How many do you get?

2.10 Define at least three sets, of which $\{\text{Alice, Diane, Eve}\}$ is a subset.

2.11 List all subsets of $\{a, b, c, d, e\}$, containing a but not containing b.

2.12 Define a set, of which both $\{1, 3, 4\}$ and $\{0, 3, 5\}$ are subsets. Find such a set with a smallest possible number of elements.

2.13 (a) Which set would you call the union of $\{a, b, c\}$, $\{a, b, d\}$ and $\{b, c, d, e\}$?

(b) Find the union of the first two sets, and then the union of this with the third. Also, find the union of the last two sets, and then the union of this with the first set. Try to formulate what you observed.

(c) Give a definition of the union of more than two sets.

2.14 Explain the connection beween the notion of the union of sets and exercise 2.2.

2.15 We form the union of a set with 5 elements and a set with 9 elements. Which of the following numbers can we get as the cardinality of the union: 4, 6, 9, 10, 14, 20?

2.16 We form the union of two sets. We know that one of them has n elements and the other has m elements. What can we infer for the cardinality of the union?

2.17 What is the intersection of

(a) the sets $\{0, 1, 3\}$ and $\{1, 2, 3\}$;

(b) the set of girls in this class and the set of boys in this class;

(c) the set of prime numbers and the set of even numbers?

2.18 We form the intersection of two sets. We know that one of them has n elements and the other has m elements. What can we infer for the cardinality of the intersection?

2.19 Prove that $|A \cup B| + |A \cap B| = |A| + |B|$.

2.20 The *symmetric difference* of two sets A and B is the set of elements that belong to exectly one of A and B.

(a) What is the symmetric difference of the set $\mathbb{Z}_+$ of non-negative integers and the set E of even integers ($E = \{\ldots -4, -2, 0, 2, 4, \ldots$ contains both negative and positive even integers).

(b) Form the symmetric difference of A ad B, to get a set C. Form the symmetric difference of A and C. What did you get? Give a proof of the answer.

2.3 The number of subsets

Now that we have introduced the notion of subsets, we can formulate our first general combinatorial problem: what is the number of all subsets of a set with n elements?

We start with trying out small numbers. It plays no role what the elements of the set are; we call them a, b, c etc. The empty set has only one subset (namely, itself). A set with a single element, say $\{a\}$, has two subsets: the set $\{a\}$ itself and the empty set $\emptyset$. A set with two elements, say $\{a, b\}$ has four subsets: $\emptyset, \{a\}, \{b\}$ and $\{a, b\}$. It takes a little more effort to list all the subsets of a set $\{a, b, c\}$ with 3 elements:

$$\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{b, c\}, \{a, c\}, \{a, b, c\}. \tag{1}$$

We can make a little table from these data:

No. of elements	0	1	2	3
No. of subsets	1	2	4	8

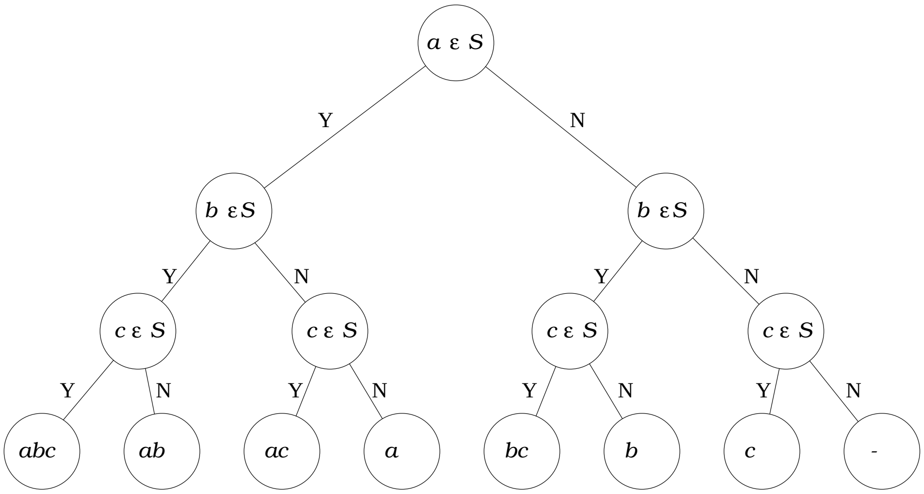

Figure 1: A decision tree for selecting a subset of $\{a, b, c\}$.

Looking at these values, we observe that the number of subsets is a power of 2: if the set has n elements, the result is 2^n, at least on these small examples.

It is not difficult to see that this is always the answer. Suppose you have to select a subset of a set A with n elements; let us call these elements $a_1, a_2, \ldots, a_n$. Then we may or may not want to include a_1, in other words, we can make two possible decisions at this point. No matter how we decided about a_1, we may or may not want to include a_2 in the subset; this means two possible decisions, and so the number of ways we can decide about a_1 and a_2 is $2 \cdot 2 = 4$. Now no matter how we decide about a_1 and a_2, we have to decide about a_3, and we can again decide in two ways. Each of these ways can be combined with each of the 4 decisions we could have made about a_1 and a_2, which makes $4 \cdot 2 = 8$ possibilities to decide about a_1, a_2 and a_3.

We can go on similarly: no matter how we decide about the first k elements, we have two possible decisions about the next, and so the number of possibilities doubles whenever we take a new element. For deciding about all the n elements of the set, we have have 2^n possibilities.

Thus we have proved the following theorem.

Theorem 2.1 *A set with n elements has 2^n subsets.*

We can illustrate the argument in the proof by the picture in Figure 1.

We read this figure as follows. We want to select a subset called S. We start from the circle on the top (called a *node*). The node contains a question: is a_1 an element of S? The two arrows going out of this node are labelled with the two possible answers to this question (Yes and No). We make a decision and follow the appropriate arrow (also called an *edge*) to the the node at the other end. This node contains the next question: is a_2 an element of S? Follow the arrow corresponding to your answer to the next node, which contains the third (and in this case last) question you have to answer to determine the subset: is a_3 an element of S? Giving an answer and following the appropriate arrow we get to a node, which contains a listing of the elements of S.

Thus to select a subset corresponds to walking down this diagram from the top to the bottom. There are just as many subsets of our set as there are nodes on the last level. Since the number of nodes doubles from level to level as we go down, the last level contains $2^3 = 8$ nodes (and if we had an n-element set, it would contain 2^n nodes).

Remark. A picture like this is called a *tree*. (This is not a mathematical definition, which we'll see later.) If you want to know why is the tree growing upside down, ask the computer scientists who introduced it, not us.

We can give another proof of theorem 2.1. Again, the answer will be made clear by asking a question about subsets. But now we don't want to select a subset; what we want is to *enumerate* subsets, which means that we want to label them with numbers $0, 1, 2, \ldots$ so that we can speak, say, about subset No. 23 of the set. In other words, we want to arrange the subsets of the set in a list and the speak about the 23rd subset on the list.

(We actually want to call the first subset of the list No. 0, the second subset on the list No. 1 etc. This is a little strange but this time it is the logicians who are to blame. In fact, you will find this quite natural and handy after a while.)

There are many ways to order the subsets of a set to form a list. A fairly natural thing to do is to start with $\emptyset$, then list all subsets with 1 elements, then list all subsets with 2 elements, etc. This is the way the list (1) is put together.

We could order the subsets as in a phone book. This method will be more transparent if we write the subsets without braces and commas. For the subsets of $\{a, b, c\}$, we get the list

$$\emptyset, a, ab, abc, ac, b, bc, c.$$

These are indeed useful and natural ways of listing all subsets. They have one shortcoming though. Imagine the list of the subsets of five elements, and ask yourself to name the 23rd subset on the list, without actually writing down the whole list. This will be difficult! Is there a way to make this easier?

Let us start with another way of denoting subsets (another *encoding* in the mathematical jargon). We illustrate it on the subsets of $\{a, b, c\}$. We look at the elements one by one, and write down a 1 if the element occurs in the subset and a 0 if it does not. Thus for the subset $\{a, c\}$, we write down 101, since a is in the subset, b is not, and c is in it again. This way every subset in "encoded" by a string of length 3, consisting of 0's and 1's. If we specify any such string, we can easily read off the subset it corresponds to. For example, the string 010 corresponds to the subset $\{b\}$, since the first 0 tells us that a is not in the subset, the 1 that follows tells us that b is in there, and the last 0 tells us that c is not there.

Now such strings consisting of 0's and 1's remind us of the *binary representation* of integers (in other words, representations in base 2). Let us recall the binary form of non-negative integers up to 10:

$$\begin{aligned}
0 &= 0_2 \\
1 &= 1_2 \\
2 &= 10_2 \\
3 &= 2 + 1 = 11_2 \\
4 &= 100_2 \\
5 &= 4 + 1 = 101_2 \\
6 &= 4 + 2 = 110_2 \\
7 &= 4 + 2 + 1 = 111_2
\end{aligned}$$

$$\begin{aligned} 8 &= 1000_2 \\ 9 &= 8+1 = 1001_2 \\ 10 &= 8+2 = 1010_2 \end{aligned}$$

(We put the subscript 2 there to remind ourselves that we are working in base 2, not 10.)

Now the binary forms of integers $0, 1, \dots, 7$ look almost as the "codes" of subsets; the difference is that the binary form of an integer always starts with a 1, and the first 4 of these integers have binary forms shorter than 3, while all codes of subsets consist of exactly 3 digits. We can make this difference disappear if we append 0's to the binary forms at their beginning, to make them all have the same length. This way we get the following correspondence:

$$\begin{array}{ccccccc} 0 & \Leftrightarrow & 0_2 & \Leftrightarrow & 000 & \Leftrightarrow & \emptyset \\ 1 & \Leftrightarrow & 1_2 & \Leftrightarrow & 001 & \Leftrightarrow & \{c\} \\ 2 & \Leftrightarrow & 10_2 & \Leftrightarrow & 010 & \Leftrightarrow & \{b\} \\ 3 & \Leftrightarrow & 11_2 & \Leftrightarrow & 011 & \Leftrightarrow & \{b,c\} \\ 4 & \Leftrightarrow & 100_2 & \Leftrightarrow & 100 & \Leftrightarrow & \{a\} \\ 5 & \Leftrightarrow & 101_2 & \Leftrightarrow & 101 & \Leftrightarrow & \{a,c\} \\ 6 & \Leftrightarrow & 110_2 & \Leftrightarrow & 110 & \Leftrightarrow & \{a,b\} \\ 7 & \Leftrightarrow & 111_2 & \Leftrightarrow & 111 & \Leftrightarrow & \{a,b,c\} \end{array}$$

So we see that the subsets of $\{a, b, c\}$ correspond to the numbers $0, 1, \dots, 7$.

What happens if we consider, more generally, subsets of a set with n elements? We can argue just like above, to get that the subsets of an n-element set correspond to integers, starting with 0, and ending with the largest integer that has only n digits in its binary representation (digits in the binary representation are usually called *bits*). Now the smallest number with $n+1$ bits is 2^n, so the subsets correspond to numbers $0, 1, 2, \dots, 2^n - 1$. It is clear that the number of these numbers in 2^n, hence the number of subsets is 2^n.

Comments. We have given two proofs of theorem 2.1. You may wonder why we needed two proofs. Certainly not because a single proof would not have given enough confidence in the truth of the statement! Unlike in a legal procedure, a mathematical proof either gives absolute certainty or else it is useless. No matter how many incomplete proofs we give, they don't add up to a single complete proof.

For that matter, we could ask you to take our word for it, and not give any proof. Later in some cases this will be necessary, when we will state theorems whose proof is too long or too involved to be included in these notes.

So why did we bother to give any proof, let alone two proofs of the same statement? The answer is that every proof reveals much more than just the bare fact stated in the theorem, and this plus may be even more valuable. For example, the first proof given above introduced the idea of breaking down the selection of a subset into independent decisions, and the representation of this idea by a tree.

The second proof introduced the idea of enumerating these subsets (labelling them with integers $0, 1, 2, \dots$). We also saw an important method of counting: we established a correspondence between the objects we wanted to count (the subsets) and some other kinds of objects that we can count easily (the numbers $0, 1, \dots, 2^n - 1$). In this correspondence

— for every subset, we had exactly one corresponding number, and

— for every number, we had exactly one corresponding subset.

A correspondence with these properties is called a *one-to-one correspondence* (or *bijection*). If we can make a one-to-one correspondence between the elements of two sets, then they have the same number of elements.

So we know that the number of subsets of a 100-element set is 2^{100}. This is a large number, but how large? It would be good to know, at least, how many digits it will have in the usual decimal form. Using computers, it would not be too hard to find the decimal form of this number, but let's try to estimate at least the order of magnitude of it.

We know that $2^3 = 8 < 10$, and hence $2^{99} < 10^{33}$. Therefore, $2^{100} < 2 \cdot 10^{33}$. Now $2 \cdot 10^{33}$ is a 2 followed by 33 zeroes; it has 34 digits, and therefore 2^{100} has at most 34 digits.

We also know that $2^{10} = 1024 > 1000 = 10^3$.[2] Hence $2^{100} > 10^{30}$, which means that 2^{100} has at least 30 digits.

This gives us a reasonably good idea of the size of 2^{100}. With a little more high school math, we can get the number of digits exactly. What does it mean that a number has exactly k digits? It means that it is between 10^{k-1} and 10^k (the lower bound is allowed, the upper is not). We want to find the value of k for which

$$10^{k-1} \leq 2^{100} < 10^k.$$

Now we can write 2^{100} in the form 10^x, only x will not be an integer: the appropriate value of x is $x = \lg 2^{100} = 100 \lg 2$. We have then

$$k - 1 \leq x < k,$$

which means that $k-1$ is the largest integer not exceeding x. Mathematicians have a name for this: it is the *integer part* or *floor* of x, and it is denoted by $\lfloor x \rfloor$. We can also say that we obtain k by rounding x down to the next integer. There is also a name for the number obtained by rounding x up to the next integer: it is called the *ceiling* of x, and denoted by $\lceil x \rceil$.

Using any scientific calculator (or table of logarithms), we see that $\lg 2 \approx 0.30103$, thus $100 \lg 2 \approx 30.103$, and rounding this down we get that $k - 1 = 30$. Thus 2^{100} has 31 digits.

2.21 Under the correspondence between numbers and subsets described above, which number correspond to subsets with 1 element?

2.22 What is the number of subsets of a set with n elements, containing a given element?

2.23 What is the number of integers with (a) at most n (decimal) digits; (b) exactly n digits?

2.24 How many bits (binary digits) does 2^{100} have if written in base 2?

2.25 Find a formula for the number of digits of 2^n.

2.4 Sequences

Motivated by the "encoding" of subsets as strings of 0's and 1's, we may want to determine the number of strings of length n composed of some other set of symbols, for example, a, b and

[2] The fact that 2^{10} is so close to 10^3 is used — or rather misused — in the name "kilobyte", which means 1024 bytes, although it should mean 1000 bytes, just like a "kilogram" means 1000 grams. Similarly, "megabyte" means 2^{20} bytes, which is close to 1 million bytes, but not exactly the same.

c. The argument we gave for the case of 0's and 1's can be carried over to this case without any essential change. We can observe that for the first element of the string, we can choose any of a, b and c, that is, we have 3 choices. No matter what we choose, there are 3 choices for the second of the string, so the number of ways to choose the first two elements is $3^2 = 9$. Going on in a similar manner, we get that the number of ways to choose the whole string is 3^n.

In fact, the number 3 has no special role here; the same argument proves the following theorem:

Theorem 2.2 *The number of strings of length n composed of k given elements is k^n.*

The following problem leads to a generalization of this question. Suppose that a database has 4 fields: the first, containing an 8-character abbreviation of an employee's name; the second, M or F for sex; the third, the birthday of the employee, in the format mm-dd-yy (disregarding the problem of not being able to distinguish emplyees born in 1880 from employees born in 1980); and the fourth, a jobcode which can be one of 13 possibilities. How many different records are possible?

The number will certainly be large. We already know from theorem 2.2 that the first field may contain $26^8 > 200,000,000,000$ names (most of these will be very difficult to pronounce, and are not likely to occur, but let's count all of them as possibilities). The second field has 2 possibile entries; the third, 36524 possible entries (the number of days in a century); the last, 13 possible entries.

Now how do we determine the number of ways these can be combined? The argument we described above can be repeated, just "3 choices" has to be replaced, in order, by "26^8 choices", "2 choices", "36524 choices" and "13 choices". We get that the answer is $26^8 \cdot 2 \cdot 36524 \cdot 13 = 198307192370919424$.

We can formulate the following generalization of theorem 2.2

Theorem 2.3 *Suppose that we want to form strings of length n so that we can use any of a given set of k_1 symbols as the first element of the string, any of a given set of k_2 symbols as the second element of the string, etc., any of a given set of k_n symbols as the last element of the string. Then the total number of strings we can form is $k_1 \cdot k_2 \cdot \ldots \cdot k_n$.*

As another special case, consider the problem: how many non-negative integers have exactly n digits (in decimal)? It is clear that the first digit can be any of 9 numbers $(1, 2, \ldots, 9)$, while the second, third, etc. digits can be any of the 10 digits. Thus we get a special case of the previous question with $k_1 = 9$ and $k_2 = k_3 = \ldots = k_n = 10$. Thus the answer is $9 \cdot 10^{n-1}$. (cf. with exercise 2.3).

2.26 Draw a tree illustrating the way we counted the number of strings of length 2 formed from the characters a, b and c, and explain how it gives the answer. Do the same for the more general problem when $n = 3$, $k_1 = 2$, $k_2 = 3$, $k_3 = 2$.

2.27 In a sport shop, there are T-shirts of 5 different colors, shorts of 4 different colors, and socks of 3 different colors. How many different uniforms can you compose from these items?

2.28 On a ticket for a succer sweepstake, you have to guess 1, 2, or X for each of 13 games. How many different ways can you fill out the ticket?

2.29 We roll a dice twice; how many different outcomes can we have (a 1 followed by a 4 is different from a 4 followed by a 1)?

2.30 We have 20 different presents that we want to distribute to 12 children. It is not required that every child gets something; it could even happen that we give all the presents to the same child. In how many ways can we distribute the presents?

2.31 We have 20 kinds of presents; this time, we have a large supply from each. We want to give presents to 12 children. Again, it is not required that every child gets something; but no child can get two copies of the same present. In how many ways can we give presents?

2.5 Permutations

During the party, we have already encountered the problem: how many ways can we seat n people on n chairs (well, we have encountered it for $n = 6$ and $n = 7$, but the question is natural enough for any n). If we imagine that the seats are numbered, then a finding a seating for these people is the same as assigning them to the numbers $1, 2, \ldots, n$ (or $0, 1, \ldots, n-1$ if we want to please the logicians). Yet another way of saying this is to order the people in a single line, or write down an (ordered) list of their names.

If we have an ordered list of n objects, and we rearrange them so that they are in another order, this is called *permuting* them, and the new order is also called a *permutation* of the objects. We also call the rearrangment that does not change anything, a *permutation* (somewhat in the spirit of calling the empty set a set).

For example, the set $\{a, b, c\}$ has the following 6 permutations:

$$abc, acb, bac, bca, cab, cba.$$

So the question is to determine the number of ways n objects can be ordered, i.e., the number of *permutations* of n objects. The solution found by the people at the party works in general: we can put any of the n people on the first place; no matter whom we choose, we have $n-1$ choices for the second. So the number of ways to fill the first two positions is $n(n-1)$. No matter how we have filled the first and second positions, there are $n-2$ choices for the third position, so the number of ways to fill the first three positions is $n(n-1)(n-2)$.

It is clear that this argument goes on like this until all positions are filled. The last but one position can be filled in two ways; the person put in the last position is determined, if the other positions are filled. Thus the number of ways to fill all positions is $n \cdot (n-1) \cdot (n-2) \cdot \ldots \cdot 2 \cdot 1$. This product is so important that we have a notation for it: $n!$ (read n factorial). In other words, $n!$ is the number of ways to order n objects. With this notation, we can state our second theorem.

Theorem 2.4 *The number of permutations of n objects in $n!$.*

Again, we can illustrate the argument above graphically (Figure 2). We start with the node on the top, which poses our first decision: whom to seat on the first chair? The 3 arrows going out correspond to the three possible answers to the question. Making a decision, we can follow one of the arrows down to the next node. This carries the next decision problem: whom to put on the second chair? The two arrows out of the node represent the two possible choices. (Note that these choices are different for different nodes on this level; what is important is that there are two arrows going out from each node.) If we make a decision and follow the

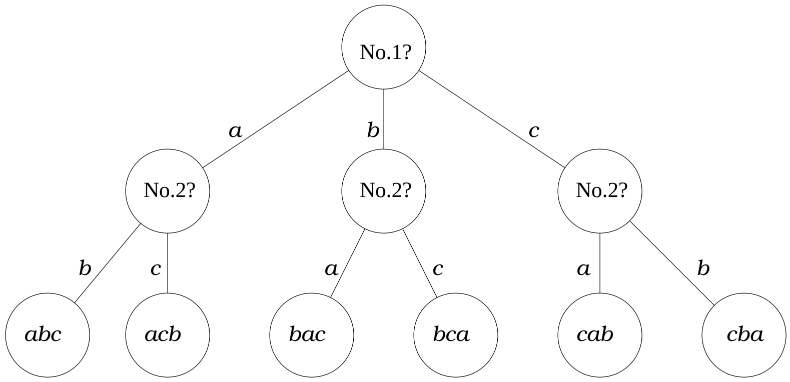

Figure 2: A decision tree for selecting a subset of $\{a, b, c\}$.

corresponding arrow to the next node, we know who sits on the third chair. The node carries the whole "seating order".

It is clear that for a set with n elements, n arrows leave the top node, and hence there are n nodes on the next level. $n-1$ arrows leave each of these, hence there are $n(n-1)$ nodes on the third level. $n-2$ arrows leave each of these, etc. The bottom level has $n!$ nodes. This shows that there are exactly $n!$ permutations.

2.32 n boys and n girls go out to dance. In how many ways can they all dance simultaneously? (We assume that only couples of different sex dance with each other.)

2.33 (a) Draw a tree for Alice's solution of enumerating the number of ways 6 people can play chess, and explain Alice's argument using the tree.

(b) Solve the problem for 8 people. Can you give a general formula for $2n$ people?

It is nice to know such a formula for the number of permutations, but often we just want to have a rough idea about how large it is. We might want to know, how many digits does 100! have? Or: which is larger, $n!$ or 2^n? In other words, does a set with n elements have more permutations or more subsets?

Let us experiment a little. For small values of n, subsets are winning: $2^1 = 2 > 1! = 1$, $2^2 = 4 > 2! = 2$, $2^3 = 8 > 3! = 6$. But then the picture changes: $2^4 = 16 < 4! = 24$, $3^5 = 32 < 5! = 120$. It is easy to see that as n increases, $n!$ grows much faster than 2^n: if we go from n to $n+1$, then 2^n grows by a factor of 2, while $n!$ grows by a factor of $n+1$.

This shows that $100! > 2^{100}$; we already know that 2^{100} has 31 digits, and hence it follows that 100! has at least 31 digits.

What *upper bound* can we give on $n!$? It is trivial that $n! < n^n$, since $n!$ is the product of n factors, each of which is at most n. (Since most of them are smaller than n, the product is in fact much smaller.) In particular, for $n = 100$, we get that $100! < 100^{100} = 10^{200}$, so 100! has at most 200 digits.

In general we know that, for $n \geq 4$,

$$2^n < n! < n^n.$$

These bounds are rather weak; for $n = 10$, the lower bound is $2^{10} = 1024$ while the upper bound is 10^{10} (i.e., ten billion).

We could also notice that $n-9$ factors in $n!$ are greater than, or equal to 10, and hence $n! \geq 10^{n-9}$. This is a much better bound for large n, but it is still far from the truth. For $n = 100$, we get that $100! \geq 10^{91}$, so it has at least 91 digits.

There is a formula that gives a very good approximation of $n!$. We state it without proof, since the proof (although not terribly difficult) needs calculus.

Theorem 2.5 [Stirling's Formula]

$$n! \sim \left(\frac{n}{e}\right)^n \sqrt{2\pi n}.$$

Here $\pi = 3.14\ldots$ is the area of the circle with unit radius, $e = 2.718\ldots$ is the basis of the natural logarithm, and $\sim$ means approximate equality in the precise sense that on the one hand

$$\frac{n!}{\left(\frac{n}{e}\right)^n \sqrt{2\pi n}} \to 1 \qquad (n \to \infty).$$

Both these funny irrational numbers e and π occur in the same formula!

So how many digits does $100!$ have? We know by Stirling's Formula that

$$100! \approx (100/e)^{100} \cdot \sqrt{200\pi}.$$

The number of digits of this number is its logarithm, in base 10, rounded up. Thus we get

$$\lg(100!) \approx 100 \lg(100/e) + 1 + \lg\sqrt{2\pi} = 157.969\ldots$$

So the number of digits in $100!$ is about 158 (actually, this is the right value).

2.34 (a) Which is larger, n or $n(n-1)/2$?
(b) Which is larger, n^2 or 2^n?

2.35 (a) Prove that $2^n > n^3$ if n is large enough.

(b) Use (a) to prove that $2^n/n^2$ becomes arbitrarily large if n is large enough.

3 Induction

3.1 The sum of odd numbers

It is time to learn one of the most important tools in discrete mathematics. We start with a question: *We add up the first n odd numbers. What do we get?*

Perhaps the best way to try to find the answer is to experiment. If we try small values of n, this is what we find:

$$\begin{aligned}
1 &= 1 \\
1+3 &= 4 \\
1+3+5 &= 9 \\
1+3+5+7 &= 16 \\
1+3+5+7+9 &= 25 \\
1+3+5+7+9+11 &= 36 \\
1+3+5+7+9+11+13 &= 49 \\
1+3+5+7+9+11+13+15 &= 64 \\
1+3+5+7+9+11+13+15+17 &= 81 \\
1+3+5+7+9+11+13+15+17+19 &= 100
\end{aligned}$$

It is easy to observe that we get squares; in fact, it seems from this examples that *the sum of the first n odd numbers is n^2.* This we have observed for the first 10 values of n; can we be sure that it is valid for all? Well, I'd say we can be reasonably sure, but not with mathematical certainty. How can we *prove* the assertion?

Consider the sum for a general n. The n-th odd number is $2n-1$ (check!), so we want ot prove that

$$1+3+\ldots+(2n-3)+(2n-1)=n^2. \tag{2}$$

If we separate the last term in this sum, we are left with the sum of the first $(n-1)$ odd numbers:

$$1+3+\ldots+(2n-3)+(2n-1)=\Big(1+3+\ldots+(2n-3)\Big)+(2n-1)$$

Now here the sum in the large paranthesis is $(n-1)^2$, so the total is

$$(n-1)^2+(2n-1)=(n^2-2n+1)+(2n-1)=n^2, \tag{3}$$

just as we wanted to prove.

Wait a minute! Aren't we using in the proof the statement that we are proving? Surely this is unfair! One could prove everything if this were allowed.

But in fact we are not quite using the same. What we were using, is the assertion about the sum of the first $n-1$ odd numbers; and we argued (in (3)) that this proves the assertion about the sum of the first n odd numbers. In other words, what we have shown is that if the assertion is true for a certain value of n, it is also true for the next.

This is enough to conclude that the assertion is true for every n. We have seen that it is true for $n=1$; hence by the above, it is also true for $n=2$ (we have seen this anyway by direct computation, but this shows that this was not even necessary: it followed from the case $n=1$).

In a similar way, the truth of the assertion for $n = 2$ implies that it is also true for $n = 3$, which in turn implies that it is true for $n = 4$, etc. If we repeat this sufficiently many times, we get the truth for any value of n.

This proof technique is called *induction* (or sometimes *mathematical induction*, to distinguish it from a notion in philosophy). It can be summarized as follows.

Suppose that we want to prove a property of positive integers. Also suppose that we can prove two facts:

(a) 1 has the property, and

(b) whenever $n - 1$ has the property, then also n has the property ($n \geq 1$).

The *principle of induction* says that if (a) and (b) are true, then every natural number has the property.

Often the best way to try to carry out an induction proof is the following. We try to prove the statement (for a general value of n), and we are allowed to use that the statement is true if n is replaced by $n - 1$. (This is called the *induction hypothesis.*) If it helps, one may also use the validity of the statement for $n - 2$, $n - 3$, etc., in general for every k such that $k < n$.

Sometimes we say that if 0 has the property, and every integer n *inherits* the property from $n - 1$, then every integer has the property. (Just like if the founding father of a family has a certain piece of property, and every new generation inherits this property from the previous generation, then the family will always have this property.)

3.1 Prove, using induction but also without it, that $n(n + 1)$ is an even number for every non-negative integer n.

3.2 Prove by induction that the sum of the first n positive integers is $n(n + 1)/2$.

3.3 Observe that the number $n(n+1)/2$ is the number of handshakes among $n+1$ people. Suppose that everyone counts only handshakes with people older than him/her (pretty snobbish, isn't it?). Who will count the largest number of handshakes? How many people count 6 handshakes?

Give a proof of the result of exercise 3.1, based on your answer to these questions.

3.4 Give a proof of exercise 3.1, based on figure 3.

3.5 Prove the following identity:

$$1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 + \ldots + (n - 1) \cdot n = \frac{(n - 1) \cdot n \cdot (n + 1)}{3}.$$

Exercise 3.1 relates to a well-known (though apocrifal) anecdote from the history of mathematics. Carl Friedrich Gauss (1777-1855), one of the greatest mathematicians of all times, was in elementary school when his teacher gave the class the task to add up the integers from 1 to 1000 (he was hoping that he would get an hour or so to relax while his students were working). To his great surprise, Gauss came up with the correct answer almost immediately. His solution was extremely simple: combine the first term with the last, you get $1 + 1000 = 1001$; combine the second term with the last but one, you get $2 + 999 = 1001$; going on in a similar way, combining the first remaining term with the last one (and then discarding them) you get 1001. The last pair added this way is $500 + 501 = 1001$. So we obtained 500 times 1001, which makes 500500. We can check this answer against the formula given in exercise 3.1: $1000 \cdot 1001/2 = 500500$.

3.6 Use the method of the little Gauss to give a third proof of the formula in exercise 3.1

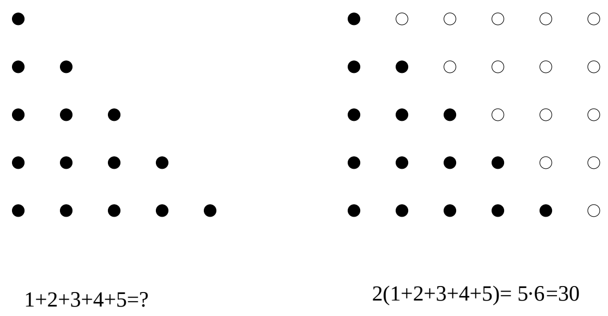

Figure 3: The sum of the first n integers

3.7 How would the little Gauss prove the formula for the sum of the first n odd numbers (2)?

3.8 Prove that the sum of the first n squares $(1 + 4 + 9 + \ldots + n^2)$ is $n(n+1)(2n+1)/6$.

3.9 Prove that the sum of the first n powers of 2 (starting with $1 = 2^0$) is $2^n - 1$.

3.2 Subset counting revisited

In chapter 2 we often relied on the convenience of saying "etc.": we described some argument that had to be repeated n times to give the result we wanted to get, but after giving the argument once or twice, we said "etc." instead of further repetition. There is nothing wrong with this, if the argument is sufficiently simple so that we can intuitively see where the repetition leads. But it would be nice to have some tool at hand which could be used instead of "etc." in cases when the outcome of the repetition is not so transparent.

The precise way of doing this is using induction, as we are going to illustrate by revisiting some of our results. First, let us give a proof of the formula for the number of subsets of an n-element set, given in Theorem 2.1 (recall that the answer is 2^n).

As the principle of induction tells us, we have to check that the assertion is true for $n = 0$. This is trivial, and we already did it. Next, we assume that $n > 0$, and that the assertion is true for sets with $n-1$ elements. Consider a set S with n elements, and fix any element $a \in S$. We want to count the subsets of S. Let us divide them into two classes: those containing a and those not containing a. We count them separately.

First, we deal with those subsets which don't contain a. If we delete a from S, we are left with a set S' with $n-1$ elements, and the subsets we are interested in are exactly the subsets of S'. By the induction hypothesis, the number of such subsets is 2^{n-1}.

Second, we consider subsets containing a. The key observation is that every such subset consists of a and a subset of S'. Conversely, if we take any subset of S', we can add a to it to get a subset of S containing a. Hence the number of subsets of S containing a is the same as the number of subsets of S', which, as we already know, is 2^{n-1}. (With the jargon we introduced before, the last piece of the argument establishes as one-to-one correspondence between those subsets of S containg a and those not containing a.)

To conclude: the total number of subsets of S is $2^{n-1} + 2^{n-1} = 2 \cdot 2^{n-1} = 2^n$. This proves Theorem 2.1 (again).

3.10 Use induction to prove Theorem 2.2 (the number of strings of length n composed of k given elements is k^n) and Theorem 2 (the number of permutations of a set with n elements is $n!$).

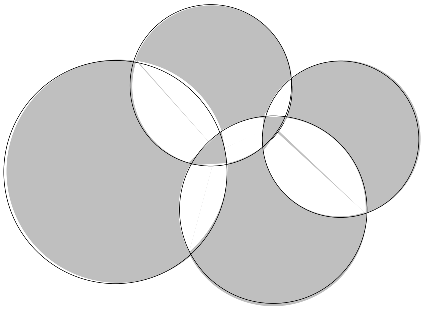

Figure 4: Two-coloring the regions formed by a set of circles

3.11 Use induction on n to prove the "handshake theorem" (the number of handshakes between n people in $n(n-1)/2$).

3.3 A few more induction proofs

We draw some circles on the plane (say, n in number). These divide the plane into a number of regions. Figure 4 shows such a set of circles, and also an "alternating" coloring of the regions with two colors; it gives a nice pattern. Now our question is: can we *always* color these regions this way? We'll show that the answer is yes; to state this more exactly:

Theorem 3.1 *The regions formed by n circles in the plane can be colored with read and blue so that any two regions that share a common boundary arc should be colored differently.*

(If two regions have only one or two boundary points in common, then they may get the same color.)

Let us first see why a direct approach does not work. We could start with coloring the outer region, say blue; then we have to color its neighbors red. Could it happen that two neighbors are at the same time neighbors of each other? Perhaps drawing some pictures and then arguing carefully about them, you can design a proof that this cannot happen. But then we have to color the neighbors of the neighbors blue again, and we would have to prove that no two of these are neighbors of each other. This could get quite complicated! And then we would have to repeat this for the neighbors of the neighbors of the neighbors...

We should find a better way to prove this and, fortunately, there is a better way! We prove the assertion by induction on n, the number of circles. If $n = 1$, then we only get two regions, and we can color one of them red, the other one blue. So let $n > 1$. Select any of the circles, say C, and forget about it for the time being. The regions formed by the remaining $n-1$ circles can be colored with red and blue so that regions that share a common boundary arc get different colors (this is just the induction hypothesis).

Now we take back the remaining circle, and change the coloring as follows: outside C, we leave the coloring as it was; inside C, we interchange red and blue (Figure 5).

It is easy to see that the coloring we obtained satisfies what we wanted. In fact, look at any small piece of arc of any of the circles. If this arc is outside C, then the two regions on its two sides were different and their colors did not change. If the arc is inside C, then again, the two regions on its both sides were differently colored, and even though their colors were switched, they are still different. Finally, the arc could be on C itself. Then the two regions on both sides of the arc were one and the same before putting C back, and so they had the

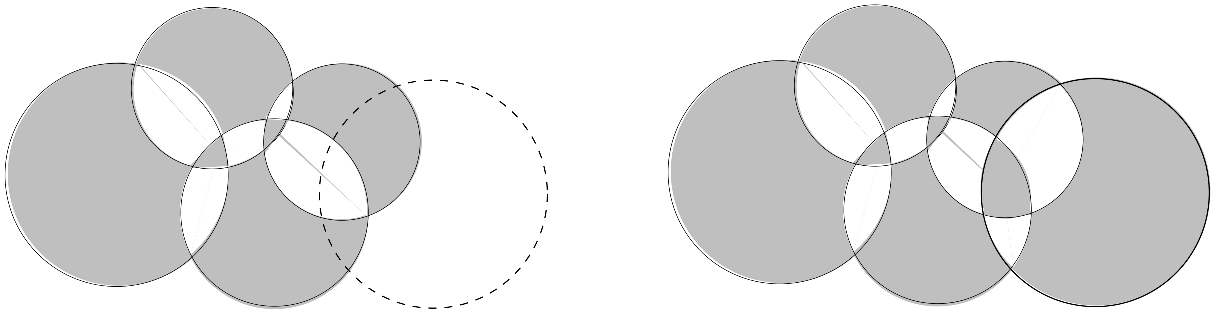

Figure 5: Adding a new circle and recoloring

same color. Now, one of them is inside C and this switched its color; the other is outside, and this did not. So after the recoloring, their colors will be different.

Thus we proved that the regions formed by n circles can be colored with two colors, provided the regions formed by $n-1$ circles can be colored with 2 colors. By the Principle of Induction, this proves the theorem.

3.12 Assume that the color of the outer region is blue. Then we can describe what the color of a particular region R is, without having to color the whole picture, as follows:

— if R lies inside an even number of circles, it will be colored blue;

— if R lies inside an odd number of circles, it will be colored red.

Prove this assertion.

3.13 (a) Prove that the regions, into which n straight lines divide the plane, are colorable with 2 colors.

(b) How could you describe what the color of a given region is?

3.14 Read carefully the following induction proof:

ASSERTION: *$n(n+1)$ is an odd number for every n.*

PROOF: *Suppose that this is true for $n-1$ in place of n; we prove it for n, using the induction hypothesis. We have*

$$n(n+1) = (n-1)n + 2n.$$

Now here $(n-1)n$ is odd by the induction hypothesis, and $2n$ is even. Hence $n(n+1)$ is the sum of an odd number and an even number, which is odd.

The assertion that we proved is obviously wrong for $n = 10$: $10 \cdot 11 = 110$ is even. What is wrong with the proof?

3.15 Read carefully the following induction proof:

ASSERTION: *If we have n lines in the plane, no two of which are parallel, then they all go through one point.*

PROOF: *The assertion is true for one line (and also for 2, since we have assumed that no two lines are parallel). Suppose that it is true for any set of $n-1$ lines. We are going to prove that it is also true for n lines, using this induction hypothesis.*

So consider a set of $S = \{a, b, c, d, \ldots\}$ of n lines in the plane, no two of which are parallel. Delete the line c, then we are left with a set S' of $n-1$ lines, and obviously no two of these

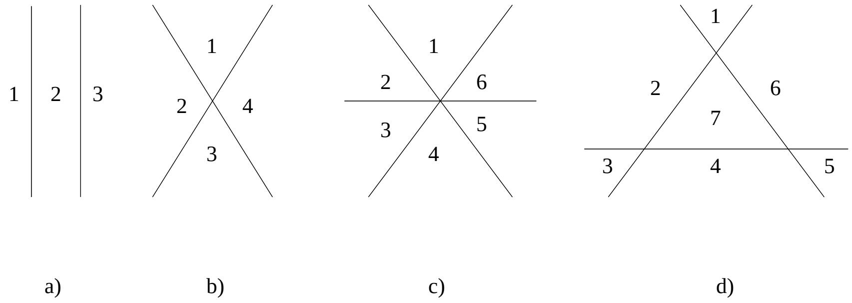

Figure 6:

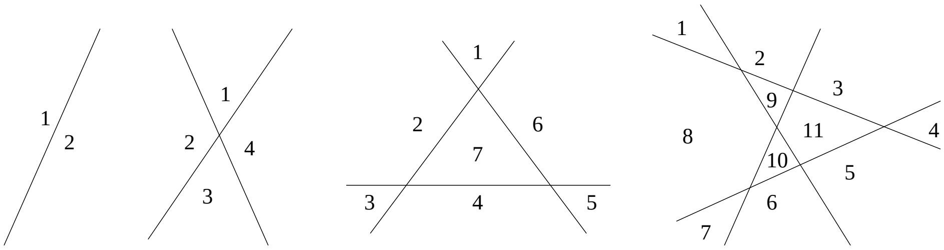

Figure 7:

are parallel. So we can apply the induction hypothesis and conclude that there is a point P such that all the lines in S' go through P. In particular, a and b go through P, and so P must be the point of intersection of a and b.

Now put c back and delete d, to get a set S'' of $n-1$ lines. Just as above, we can use the induction hypothesis to conclude that these lines go through the same point P'; but just like above, P' must be the point of intersection of a and b. Thus $P' = P$. But then we see that c goes through P. The other lines also go through P (by the choice of P), and so all the n lines go thorugh P.

But the assertion we proved is clearly wrong; where is the error?

3.4 Counting regions

Let us draw n lines in the plane. These lines divide the plane into some number of regions. How many regions do we get?

A first thing to notice is that this question does not have a single answer. For example, if we draw two lines, we get 3 regions if the two are parallel, and 4 regions if they are not.

OK, let us assume that no two of the lines are parallel; then 2 lines always give us 4 regions. But if we go on to three lines, we get 6 regions if the lines go through one point, and 7 regions, if they do not (Figure 6).

OK, let us also exclude this, and assume that no 3 lines go through the same point. One might expect that the next unpleasant example comes with 4 lines, but if you experiment with drawing 4 lines in the plane, with no two parallel and no three going threw the same point, then you invariably get 11 regions (Figure 7). In fact, we'll hvae a similar experience for any number of lines.

A set of lines in the plane such that no two are parallel and no three go through the same point is said to be *in general position*. If we choose the lines "randomly" then accidents like two being parallel or three going through the same point will be very unlikely, so our assumption that the lines are in general position is quite natural.

Even if we accept that the number of regions is always the same for a given number of lines, the question still remains: what is this number? Let us collect our data in a little table (including also the observation that 0 lines divide the plane into 1 region, and 1 line divides the plane into 2):

0	1	2	3	4
1	2	4	7	11

Staring at this table for a while, we observe that each number in the second row is the sum of the number above it and the number before it. This suggests a rule: the n-th entry is n plus the previous entry. In other words: *If we have a set of $n-1$ lines in the plane in general position, and add a new line (preserving general position), then the number of regions increses by n.*

Let us prove this assertion. How does the new line increase the number of regions? By cutting some of them into two. The number of additional regions is just the same as the number of regions intersected.

So, how many regions does the new line intersect? At a first glance, this is not easy to answer, since the new line can intersect very different sets of regions, depending on where we place it. But imagine to walk along the new line, starting from very far. We get to a new region every time we cross a line. So the number of regions the new line intersects is one larger than the number of crossing points on the new line with other lines.

Now the new line crosses every other line (since no two lines are parallel), and it crosses them in different points (since no three lines go through the same point). Hence during our walk, we see $n-1$ crossing points. So we see n different regions. This proves that our observation about the table is true for every n.

We are not done yet; what does this give for the number of regions? We start with 1 region for 0 lines, and then add to it $1, 2, 3, \ldots, n$. This way we get

$$1 + (1 + 2 + 3 + \ldots + n) = 1 + \frac{n(n+1)}{2}.$$

Thus we have proved:

Theorem 3.2 *A set of n lines in general position in the plane divides the plane into $1+n(n+1)/2$ regions.*

3.16 Describe a proof of Theorem 3.2 using induction on the number of lines.

Let us give another proof of Theorem 3.2; this time, we will not use induction, but rather try to relate the number of regions to other combinatorial problems. One gets a hint from writing the number in the form $1 + n + \binom{n}{2}$.

Assume that the lines are drawn on a vertical blackboard (Figure 8), which is large enough so that all the intersection points appear on it. We also assume that no line is horizontal (else, we tilt the picture a little), and that in fact every line intersects the bottom edge of the blackboard (the blackboard is very long).

Now consider the lowest point in each region. Each region has only one lowest point, since the bordering lines are not horizontal. This lowest point is then an intersection point of two of our lines, or the intersection point of line with the lower edge of the blackboard, or the lower

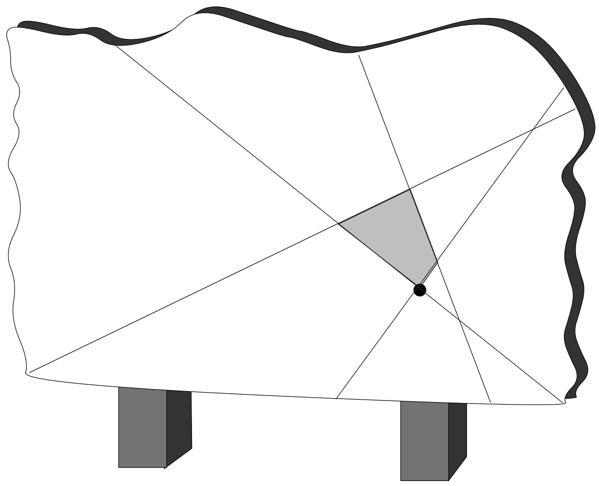

Figure 8:

left corner of the blackboard. Furthermore, each of these points is the lowest point of one and only one region. For example, if we consider any intersection point of two lines, then we see that four regions meet at this point, and the point is the lowest point of exactly one of them.

Thus the number of lowest points is the same as the number of intersection points of the lines, plus the number of intersection points between lines and the lower edge of the blackboard, plus one. Since any two lines intersect, and these intersection points are all different (this is where we use that the lines are in general position), the number of such lowest points is $\binom{n}{2}+n+1$.

4 Counting subsets

4.1 The number of ordered subsets

At a competition of 100 athletes, only the order of the first 10 is recorded. How many different outcomes does the competition have?

This question can be answered along the lines of the arguments we have seen. The first place can be won by any of the athletes; no matter who wins, there are 99 possible second place winners, so the first two prizes can go $100 \cdot 99$ ways. Given the first two, there are 98 athletes who can be third, etc. So the answer is $100 \cdot 99 \cdot \ldots \cdot 91$.

4.1 Illustrate this argument by a tree.

4.2 Suppose that we record the order of all 100 athletes.
(a) How many different outcomes can we have then?
(b) How many of these give the same for the first 10 places?
(c) Show that the result above for the number of possible outcomes for the first 10 places can be also obtained using (a) and (b).

There is nothing special about the numbers 100 and 10 in the problem above; we could carry out the same for n athletes with the first k places recorded.

To give a more mathematical form to the result, we can replace the athletes by any set of size n. The list of the first k places is given by a sequence of k elements of n, which all have to be different. We may also view this as selecting a subset of the athletes with k elements, and then ordering them. Thus we have the following theorem.

Theorem 4.1 *The number of ordered k-subsets of an n-set is $n(n-1)\ldots(n-k+1)$.*

(Note that if we start with n and count down k numbers, the last one will be $n-k+1$.)

4.3 If you generalize the solution of exercise 4.1, you get the answer in the form

$$\frac{n!}{(n-k)!}$$

Check that this is the same number as given in theorem 4.1.

4.4 Explain the similarity and the difference between the counting questions answered by theorem 4.1 and theorem 2.2.

4.2 The number of subsets of a given size

From here, we can easily derive one of the most important counting results.

Theorem 4.2 *The number of k-subsets of an n-set is*

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

Recall that if we count *ordered* subsets, we get $n(n-1)\dots(n-k+1) = n!/(n-k)!$, by Theorem 4.1. Of course, if we want to know the number of *unordered* subsets, then we have overcounted; every subset was counted exactly $k!$ times (with every possible ordering of its elements). So we have to divide this number by $k!$ to get the number of subsets with k elements (without ordering).

The number of k-subsets of an n-set is such an important quantity that one has a separate notation for it: $\binom{n}{k}$ (read: 'n choose k'). Thus

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

Thus the number of different lottery tickets in $\binom{90}{5}$, the number of handshakes is $\binom{7}{2}$ etc.

4.5 Which problems discussed during the party were special cases of theorem 4.2?

4.6 Tabulate the values of $\binom{n}{k}$ for $n, k \le 5$.

In the following exercises, try to prove the identities by using the formula in theorem 4.2, and also without computation, by explaining both sides of the equation as the result of a counting problem.

4.7 Prove that $\binom{n}{2} + \binom{n+1}{2} = n^2$.

4.8 (a) Prove that $\binom{90}{5} = \binom{89}{5} + \binom{89}{4}$.

(b) Formulate and prove a general identity based on this.

4.9 Prove that $\binom{n}{k} = \binom{n}{n-k}$.

4.10 Prove that

$$1 + \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{n-1} + \binom{n}{n} = 2^n.$$

4.11 Prove that for $0 < c \le b \le a$,

$$\binom{a}{b}\binom{b}{c} = \binom{a}{a-c}\binom{a-c}{b-c}$$

4.3 The Binomial Theorem

The numbers $\binom{n}{k}$ also have a name, *binomial coefficients*, which comes from a very important formula in algebra involving them. We are now going to discuss this theorem.

The issue is to compute powers of the simple algebraic expression $(x+y)$. We start with small examples:

$$(x+y)^2 = x^2 + 2xy + y^2,$$

$$(x+y)^3 = (x+y)\cdot(x+y)^2 = (x+y)\cdot(x^2+2xy+y^2) = x^3 + 3x^2y + 3xy^2 + y^3,$$

and, going on like this,

$$(x+y)^4 = (x+y)\cdot(x+y)^3 = x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4.$$

You may have noticed that the coefficients you get are the numbers that we have seen, e.g. in exercise 4.2, as numbers $\binom{n}{k}$. Let us make this observation precise. We illustrate the argument for the next value of n, namely $n = 5$, but it works in general.

Think of expanding

$$(x+y)^5 = (x+y)(x+y)(x+y)(x+y)(x+y)$$

so that we get rid of all paranthesis. We get each term in the expansion by selecting one of the two terms in each factor, and multiplying them. If we choose x, say, 2 times then we choose y 3 times, and we get x^2y^3. How many times do we get this same term? Clearly as many times as the number of ways to select the two factors that supply x (the remaining factors supply y). Thus we have to choose two factors out of 5, which can be done in $\binom{5}{2}$ ways.

Hence the expansion of $(x+y)^5$ looks like this:

$$(x+y)^5 = \binom{5}{0}y^5 + \binom{5}{1}xy^4 + \binom{5}{2}x^2y^3 + \binom{5}{3}x^3y^2 + \binom{5}{4}x^4y + \binom{5}{5}x^5.$$

We can apply this argument in general to obtain

Theorem 4.3 (The Binomial Theorem) *The coefficient of x^ky^{n-k} in the expansion of $(x+y)^n$ is $\binom{n}{k}$. In ither words, we have the identity:*

$$(x+y)^n = y^n + \binom{n}{1}x^{n-1}y + \binom{n}{2}x^{n-2}y^2 + \ldots + \binom{n}{n-1}x^{n-1}y + \binom{n}{n}x^n.$$

This important theorem is called the Binomial Theorem; the name comes from the Greek word *binome* for an expression consisting of two terms, in this case, $x + y$. The appearence of the numbers $\binom{n}{k}$ in this theorem is the source of their name: *binomial coefficients.*

The Binomial Theorem can be applied in many ways to get identities concerning binomial coefficients. For example, let us substitute $x = y = 1$, then we get

$$2^n = \binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \ldots + \binom{n}{n-1} + \binom{n}{n}. \tag{4}$$

Later on we are going to see trickier applications of this idea. For the time being, another twist on it is contained in the next exercise.

4.12 Give a proof of the Binomial Theorem by inducion, based on exercise 4.2.

4.13 (a) Prove the identity

$$\binom{n}{0} - \binom{n}{1} + \binom{n}{2} - \binom{n}{3} \ldots = 0$$

(The sum ends with $\binom{n}{n} = 1$, with the last depending on the parity of n.)

(b) This identity is obvious if n is odd. Why?

4.14 Prove identity 4, using a combinatorial interpretation of the two sides (recall exercise 4.2).

4.4 Distributing presents

Suppose we have n different presents, which we want to distribute to k children. For some reason, we are told how many presents should each child get; so Adam should get n_{Adam} presents, Barbara, n_{Barbara} presents etc. In a mathematically convenient (though not very friendly) way, we call the children $1, 2, \ldots, k$; thus we are given the numbers (non-negative integers) $n_1, n_2, \ldots, n_k$. We assume that $n_1 + n_2 + \ldots + n_k = n$, else there is no way to distribute the presents.

The question is, of course, how many ways can these presents be distributed?

We can organize the distribution of presents as follows. We lay out the presents in a single row of length n. The first child comes and picks up the first n_1 presents, starting from the left. Then the second comes, and picks up the next n_2; then the third picks up the next n_3 presents etc. Child No. k gets the last n_k presents.

It is clear that we can determine who gets what by choosing the order in which the presents are laid out. There are $n!$ ways to order the presents. But, of course, the number $n!$ overcounts the number of ways to distribute the presents, since many of these orderings lead to the same results (that is, every child gets the same set of presents). The question is, how many?

So let us start with a given distribution of presents, and let's ask the children to lay out the presents for us, nicely in a row, starting with the first child, then continuing with the second, third, etc. This way we get back *one* possible ordering that leads to the current distribution. The first child can lay out his presents in $n_1!$ possible orders; no matter which order he choses, the second child can lay out her presents in $n_2!$ possible ways, etc. So the number of ways the presents can be laid out (given the distribution of the presents to the children) is a product of factorials:

$$n_1! \cdot n_2! \cdot \ldots \cdot n_k!\,.$$

Thus the number of ways of distributing the presents is

$$\frac{n!}{n_1! n_2! \ldots n_k!}\,.$$

4.15 We can describe the procedure of distributing the presents as follows. First, we select n_1 presents and give them to the first child. This can be done in $\binom{n}{n_1}$ ways. Then we select n_2 presents from the remaining $n - n_1$ and give them to the second child, etc. Complete this argument and show that it leads to the same result as the previous one.

4.16 The following special cases should be familiar from previous problems and theorems. Explain why.

(a) $n = k, n_1 = n_2 = \ldots = n_k$;

(b) $n_1 = n_2 = \ldots = n_{k-1} = 1, n_k = n - k + 1$;

(c) $k = 2$;

(d) $k = 3, n = 6, n_1 = n_2 = n_3 = 2$.

4.17 (a) How many ways can you place n rooks on a chessboard so that no two attack each other (Figure 9)? We assume that the rooks are identical, so e.g. interchanging two rooks does not count as a separate placement.

(b) How many ways can you do this if you have 4 black and 4 white rooks?

(c) How many ways can you do this if all the 8 rooks are different?

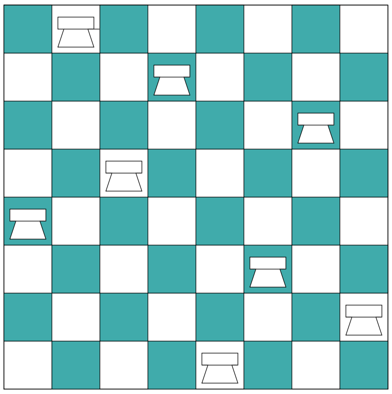

Figure 9: Placing 8 non-attacking rooks on a chessboard

4.5 Anagrams

Quite probably you have played with anagrams. One selects a word (say, COMBINATORICS) and tries to compose from its letters meaningful, often funny words or expressions.

How many anagrams can you build from a given word? If you try to answer this question by playing around with the letters, you will realize that the question is badly posed; it is difficult to draw the line between meaningful and non-meaningful anagrams. For example, it could easily happen that A CROC BIT SIMON. And it may be true that Napoleon always wanted a TOMB IN CORSICA. It is questionable, but certainly grammatically correct, to assert that COB IS ROMANTIC. Some universities may have a course on MAC IN ROBOTICS.

But one would have to write a book to introduce an exciting character, ROBIN COSMICAT, who enforces a COSMIC RIOT BAN, while appealing TO COSMIC BRAIN.

And it would be terribly difficult to explain an anagram like MTBIRASCIONOC.

To avoid this contraversy, let's accept everything, i.e., we don't require the anagram to be meaningful (or even pronouncible). Of course, the production of anagrams becomes then uninteresting; but at least we can tell how many of them are there!

4.18 How many anagrams can you make from the word COMBINATORICS?

4.19 Which word gives rise to more anagrams: COMBINATORICS or COMBINATORICA? (The latter is the Latin name of the subject.)

4.20 Which word with 13 letters gives rise to the most anagrams? Which word gives rise to the least?

So let's see the general answer to the question of counting anagrams. If you have solved the problems above, it should be clear that the number of anagrams n-letter word depends on how many times letters of the word are repeated. So suppose that the word contains letter No. 1 n_1 times, letter No. 2 n_2 times, etc., letter No. k n_k times. Clearly, $n_1 + n_2 + \ldots + n_k = n$.

Now to form an anagram, we have to select n_1 positions for letter No. 1, n_2 positions for letter No. 2, etc., n_k positions fro letter No. 3. Having formulated it this way, we can see that this is nothing but the question of distributing n presents to k children, when it is prescribed how many presents each child gets. Thus we know from the previous section that the answer is

$$\frac{n}{n_1!n_2!\ldots n_k!}.$$

4.21 It is clear that STATUS and LETTER have the same number of anagrams (in fact, $6!/(2!2!) = 180$). We say that these words are "essentially the same" (at least as far as counting anagrams goes): they have two letters repeated twice and two letters occuring only once.

(a) How many 6-letter words are there? (As before, the words don't have to be meaningful. The alphabet has 26 letters.)

(b) How many words with 6 letters are "essentially the same" as the word LETTER?

(c) How many "essentially different" 6-letter words are there?

(d) Try to find a general answer to question (c) (that is, how many "essentially different" words are there on n letters?). If you can't find it, read the following section and return to this exercise after it.

4.6 Distributing money

Instead of distributing presents, let's distribute money. Let us formulate the question in general: we have n pennies that we want to distribute among k kids. Each child must get at least one penny (and, of course, an integer number of pennies). How many ways can we distribute the money?

Before answering this question, we must clarify the difference between distributing money and distributing presents. If you are distributing presents, you have to decide not only how many presents each child gets, but also *which* are these presents. If you are distributing money, only the quantity matters. In other words, the presents are *distinguishable* while the pennies are not. (A question like in section 4.4, where we specify in advence how many presents does a given child get, would be trivial for money: there is only one way to distribute n pennies so that the first child gets n_1, the second child gets n_2, etc.)

Even though the problem is quite different from the distribution of presents, we can solve it by imagining a similar distribution method. We line up the pennies (it does not matter in which order, they are all alike), and then let child No. 1 begin to pick them up from left to right. After a while we stop him and let the second child pick up pennies, etc. (Figure 10) *The distribution of the money is determined by specifying where to start with a new child.*

Now there are $n-1$ points (between consecutive pennies) where we can let a new child in, and we have to select $k-1$ of them (since the first child alsways starts at the beginning, we have no choice there). Thus we have to select a $(k-1)$-element subset from an $(n-1)$-element set. The number of possibilities to do so is $\binom{n-1}{k-1}$.

To sum up, we get

Theorem 4.4 *The number of ways to distribute n identical pennies to k children, so that each child gets at least one, is $\binom{n-1}{k-1}$.*

It is quite surprising that the binomial coefficients give the answer here, in a quite non-trivial and unexpected way.

Let's also discuss the natural (though unfair) modification of this question, where we also allow distributions in which some children get no money at all; we consider even giving all the money to one child. With the following trick, we can reduce the problem of counting such distribnutions to the problem we just solved: we borrow 1 penny from each child, and the distribute the whole amount (i.e., $n+k$ pennies) to the children so that each child gets at least one penny. This way every child gets back the money we borrowed from him or her, and

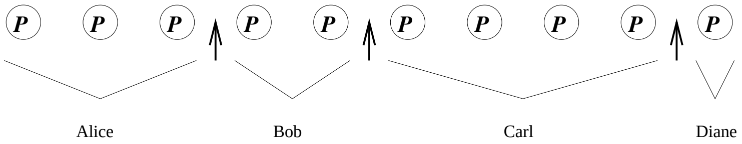

Figure 10: How to distribute n pennies to k children?

the lucky ones get some more. The "more" is exactly n pennies distributed to k children. We already know that the number of ways to distribute $n+k$ pennies to k children so that each child gets at least one penny is $\binom{n+k-1}{k-1}$. So we have

Theorem 4.5 *The number of ways to distribute n identical pennies to k children is $\binom{n+k-1}{k-1}$.*

4.22 In how many ways can you distribute n pennies to k children, if each child is supposed to get at least 2?

4.23 We distribute n pennies to k boys and ℓ girls, so that (to be really unfair) we require that each of the girls gets at least one penny. In how many ways can we do this?

4.24 k earls play cards. Originally, they all have p pennies. At the end of the game, they count how much money they have. They do not borrow from each other, so that they cannot loose more than their p pennies. How many possible results are there?

5 Pascal's Triangle

To study various properties of binomial coefficients, the following picture is very useful. We arrange all binomial coefficients into a triangular scheme: in the "zeroeth" row we put $\binom{0}{0}$, in the first row, we put $\binom{1}{0}$ and $\binom{1}{1}$, in the third row, $\binom{2}{0}$, $\binom{2}{1}$ and $\binom{2}{2}$ etc. In general, the n-th row contains the numbers $\binom{n}{0}, \binom{n}{1}, \dots, \binom{n}{n}$. We shift these rows so that their midpoints match; this way we get a pyramid-like scheme, called the Pascal Triangle (named after the French mathematician and philosopher Blaise Pascal, 1623-1662). The Figure below shows only a finite piece of the Pascal Triangle.

$$
\begin{array}{ccccccccccc}
 & & & & & \binom{0}{0} & & & & & \\
 & & & & \binom{1}{0} & & \binom{1}{1} & & & & \\
 & & & \binom{2}{0} & & \binom{2}{1} & & \binom{2}{1} & & & \\
 & & \binom{3}{0} & & \binom{3}{1} & & \binom{3}{2} & & \binom{3}{3} & & \\
 & \binom{4}{0} & & \binom{4}{1} & & \binom{4}{2} & & \binom{4}{3} & & \binom{4}{4} & \\
\binom{5}{0} & & \binom{5}{1} & & \binom{5}{2} & & \binom{5}{3} & & \binom{5}{4} & & \binom{5}{5}
\end{array}
$$

We can replace each binomial coefficient by its numerical value, to get another version of Pascal's Triangle.

$$
\begin{array}{ccccccccccccccccc}
 & & & & & & & & 1 & & & & & & & & \\
 & & & & & & & 1 & & 1 & & & & & & & \\
 & & & & & & 1 & & 2 & & 1 & & & & & & \\
 & & & & & 1 & & 3 & & 3 & & 1 & & & & & \\
 & & & & 1 & & 4 & & 6 & & 4 & & 1 & & & & \\
 & & & 1 & & 5 & & 10 & & 10 & & 5 & & 1 & & & \\
 & & 1 & & 6 & & 15 & & 20 & & 15 & & 6 & & 1 & & \\
 & 1 & & 7 & & 21 & & 35 & & 35 & & 21 & & 7 & & 1 & \\
1 & & 8 & & 28 & & 56 & & 70 & & 56 & & 28 & & 8 & & 1
\end{array}
$$

5.1 Prove that the Pascal Triangle is symmetric with respect to the vertical line through its apex.

5.2 Prove that each row in the Pascal Triangle starts and ends with 1.

5.1 Identities in the Pascal Triangle

Looking at the Pascal Triangle, it is not hard to notice its most important property: every number in it (other than the 1's on the boundary) is the sum of the two numbers immediately above it. This in fact is a property of the binomial coefficients you have already met: it translates into the relation

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k} \tag{5}$$

(cf. exercise 4.2).

This property of the Pascal Triangle enables us to generate the triangle very fast, building it up row by row, using (5). It also gives us a tool to prove many properties of the binomial coefficients, as we shall see.

As a first application, let us give a new solution of exercise 4.3. There the task was to prove the identity

$$\binom{n}{0} - \binom{n}{1} + \binom{n}{2} - \binom{n}{3} \ldots + (-1)^n \binom{n}{n} = 0,$$

using the binomial theorem. Now we give a proof based on (5): we can replace $\binom{n}{0}$ by $\binom{n-1}{0}$ (both are just 1), $\binom{n}{1}$ by $\binom{n-1}{0} + \binom{n-1}{1}$, $\binom{n}{2}$ by $\binom{n-1}{1} + \binom{n-1}{2}$, etc. Thus we get the sum

$$\binom{n-1}{0} - \left[\binom{n-1}{0} + \binom{n-1}{1}\right] + \left[\binom{n-1}{1} + \binom{n-1}{2}\right] - \left[\binom{n-1}{2} + \binom{n-1}{3}\right] + \ldots$$

which is clearly 0, since the second term in each bracket cancels with the first term of the next.

This method gives more than just a new proof of an identity we already know. What do we get if we start the same way, adding and subtracting binomial coefficients alternatingly, but stop earlier? In formula, we take

$$\binom{n}{0} - \binom{n}{1} + \binom{n}{2} - \binom{n}{3} \ldots + (-1)^k \binom{n}{k}.$$

If we do the same trick as above, we get

$$\binom{n-1}{0} - \left[\binom{n-1}{0} + \binom{n-1}{1}\right] + \left[\binom{n-1}{1} + \binom{n-1}{2}\right] - \ldots (-1)^k \left[\binom{n-1}{k-1} + \binom{n-1}{k}\right].$$

Here again every term cancels except the last one; so the result is $(-1)^k \binom{n-1}{k}$.

There are many other surprising relations satisfied by the numbers in the Pascal Triangle. For example, let's ask: what is the sum of *squares* of elements in each row?

Let's experiment, by computing the sum of squares of elements in the the first few rows:

$$\begin{aligned} 1^2 &= 1, \\ 1^2 + 1^2 &= 2, \\ 1^2 + 2^2 + 1^2 &= 6, \\ 1^2 + 3^2 + 3^2 + 1^2 &= 20, \\ 1^2 + 4^2 + 6^2 + 4^2 + 1^2 &= 70. \end{aligned}$$

We may recognize these numbers as the numbers in the middle column of the Pascal triangle. Of course, only every second row contains an entry in the middle column, so the last value above, the sum of squares in row No. 4, is the middle element in row No. 8. So the examples above suggest the following identity:

$$\binom{n}{0}^2 + \binom{n}{1}^2 + \binom{n}{2}^2 + \ldots + \binom{n}{n-1}^2 + \binom{n}{n}^2 = \binom{2n}{n}. \tag{6}$$

Of course, the few experiments above do not prove that this identity always holds, so we need a proof.

We will give an interpretation of both sides of the identity as the result of a counting problem; it will turn out that they count the same things, so they are equal. It is obvious what the right hand side counts: the number of subsets of size n of a set of size $2n$. It will be convenient to choose, as our $2n$-element set, the set $S = \{1, 2, \dots, 2n\}$.

The combinatorial interpretation of the left hand side is not so easy. Consider a typical term, say $\binom{n}{k}^2$. We claim that this is the number of those n-element subsets of $\{1, 2, \dots, 2n\}$ that contain exactly k elements from $\{1, 2, \dots, n\}$ (the first half of our set S). In fact, how do we choose such an n-element subset of S? We choose k elements from $\{1, 2, \dots, n\}$ and then $n-k$ elements from $\{n+1, n+2, \dots, 2n\}$. The first can be done in $\binom{n}{k}$ ways; no matter which k-element subset of $\{1, 2, \dots, n\}$ we selected, we have $\binom{n}{n-k}$ ways of choose the other part. Thus the number of ways to choose an n-element subset of S having k elements from $\{1, 2, \dots, n\}$ is

$$\binom{n}{k} \cdot \binom{n}{n-k} = \binom{n}{k}^2$$

(by the symmetry of the Pascal Triangle).

Now to get the total number number of n-element subsets of S, we have to sum these numbers for all values of $k = 0, 1, \dots, n$. This proves identity (6).

5.3 Give a proof of the formula in exercise 4.2

$$1 + \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{n-1} + \binom{n}{n} = 2^n$$

along the same lines. (One could expect that, similarly as for the "alternating" sum, we could get a nice formula for the sum obtained by stopping earlier, like $\binom{n}{0} + \binom{n}{1} + \dots + \binom{n}{k}$. But this is not the case: no simpler expression is known for this sum in general.)

5.4 By the Binomial Theorem, the right hand side in identity (6) is the coefficient of $x^n y^n$ in the expansion of $(x+y)^{2n}$. Write $(x+y)^{2n}$ in the form $(x+y)^n(x+y)^n$, expand both factors $(x+y)^n$ using the binomial theorem, and the try to figure out the coefficient of $x^n y^n$ in the product. Show that this gives another proof of identity (6).

5.5 Prove the following identity:

$$\binom{n}{0}\binom{m}{k} + \binom{n}{1}\binom{m}{k-1} + \binom{n}{2}\binom{m}{k-2} + \dots + \binom{n}{k-1}\binom{m}{1} + \binom{n}{k}\binom{m}{0} = \binom{n+m}{k}.$$

You can use a combinatorial interpretation of both sides, similarly as in the proof of (6) above, or the Binomial Theorem as in the previous exercise.

Here is another relation between the numbers in the Pascal Triangle. Let us start with the last element in any row, and sum the elements moving down diagonally to the left. For example, starting with the last element in the second row, we get

$$\begin{aligned} 1 &= 1, \\ 1+3 &= 4, \\ 1+3+6 &= 10, \\ 1+3+6+10 &= 20. \end{aligned}$$

This numbers are just the numbers in the next skew line of the table! If we want to put this in a formula, we get

$$\binom{n}{0} + \binom{n+1}{1} + \binom{n+2}{2} + \ldots + \binom{n+k}{k} = \binom{n+k+1}{k}. \tag{7}$$

To *prove* this identity, we use induction on k. If $k = 0$, the identity just says $1 = 1$, so it is trivially true. (We can check it also for $k = 1$, even though this is not necessary. Anyway, it says $1 + n = n + 1$.)

So suppose that the identity (7) is true for a given value of k, and we want to prove that it also holds for $k + 1$ in place of k. In other words, we want to prove that

$$\binom{n}{0} + \binom{n+1}{1} + \binom{n+2}{2} + \ldots + \binom{n+k}{k} + \binom{n}{k+1} = \binom{n+k+2}{k+1}.$$

Here the sum of the first k terms on the left hand side is $\binom{n+k+1}{k}$ by the induction hypothesis, and so the left hand side is equal to

$$\binom{n+k+1}{k} + \binom{n+k+1}{k+1}.$$

But this is indeed equal to $\displaystyle\binom{n+k+2}{k+1}$ by the fundamental property of the Pascal Triangle. This completes the proof by induction.

5.6 Suppose that you want to choose a $(k + 1)$-element subset of the $(n + k + 1)$-element set $\{1, 2, \ldots, n + k + 1\}$. You decide to do this by choosing first the largest element, then the rest. Show that counting the number of ways to choose the subset this way, you get a combinatorial proof of identity (7).

5.2 A bird's eye view at the Pascal Triangle

Let's imagine that we are looking at the Pascal Triangle from a distance. Or, to put it differently, we are not interested in the exact numerical value of the entries, but rather in their order of magnitude, rise and fall, and other global properties. The first such property of the Pascal Triangle is its symmetry (with respect to the vertical line through its apex), which we already know.

Another property one observes is that along any row, the entries increase until the middle, and then decrease. If n is even, there is a unique middle element in the n-th row, and this is the largest; if n is odd, then there are two equal middle elements, which are largest.

So let us *prove* that the entries increase until the middle (then they begin to decrease by the symmetry of the table). We want to compare two consecutive entries:

$$\binom{n}{k} \ ? \ \binom{n}{k+1}.$$

If we use formula 4.2, we can write this as

$$\frac{n(n-1)\dots(n-k+1)}{k(k-1)\dots 1} \ ? \ \frac{n(n-1)\dots(n-k)}{(k+1)k\dots 1}.$$

There are a lot of common factors on both sides with which we can simplify. We get the really simple comparison

$$1 \ ? \ \frac{n-k}{k+1}.$$

Rearranging, we get

$$k \ ? \ \frac{n-1}{2}.$$

So if $k < (n-1)/2$, then $\binom{n}{k} < \binom{n}{k+1}$; if $k = (n-1)/2$, then $\binom{n}{k} = \binom{n}{k+1}$ (this is the case of the two entries in the middle if n is odd); and if $k > (n-1)/2$, then $\binom{n}{k} > \binom{n}{k+1}$.

It will be useful later that this computation also describes by *how much* consecutive elements increase or decrease. If we start from the left, the second entry (n) is larger by a factor of n than the first; the third ($n(n-1)/2$) is larger by a factor of $(n-1)/2$ than the second. In general,

$$\frac{\binom{n}{k+1}}{\binom{n}{k}} = \frac{n-k}{k+1}. \tag{8}$$

5.7 For which values of n and k is $\binom{n}{k+1}$ twice the previous entry in the Pascal Triangle?

5.8 Instead of the ratio, look at the difference of two consecutive entries in the Pascal triangle:

$$\binom{n}{k+1} - \binom{n}{k}.$$

For which value of k is this difference largest?

We know that each row of the Pascal Triangle is symmetric. We also know that the entries start with 1, raise to the middle, and then they fall back to 1. Can we say more about their shape?

Figure 11 shows the graph of the numbers $\binom{n}{k}$ $(k = 0, 1, \dots, n)$ for the values $n = 10$ and $n = 100$. We can make several further observations.

— First, the largest number gets very large.

— Second, not only do these numbers increase to the middle and then they decrase, but that the middle ones are substantially larger than those at the beginning and end. For $n = 100$, the figure only shows the range $\binom{100}{20}, \binom{100}{21}, \dots, \binom{100}{80}$; the numbers outside this range are so small compared to the largest that they are negligible.

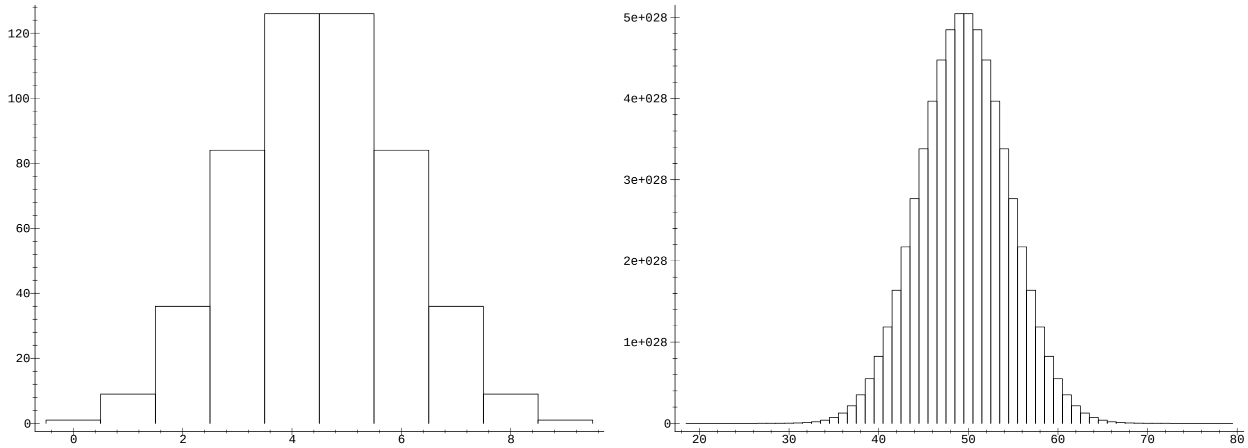

Figure 11: Graph of the n-th row of the Pascal Triangle, for $n = 10$ and 100.

— Third, we can observe that the shape of the graph is quite similar for different values of n.

Let's look more carefully at these observations. For the discussions that follow, we shall assume that n is even (for odd values of n, the results would be quite similar, just one would have to word them differently). If n is even, then we know that the largest entry in the n-th row is the middle number $\binom{n}{n/2}$, and all other entries are smaller.

How large is the largest number in the n-th row of the Pascal Triangle? We know immediately an upper bound on this number:

$$\binom{n}{n/2} < 2^n,$$

since 2^n is the sum of all entries in the row. It only takes a little more sophistication to get a lower bound:

$$\binom{n}{n/2} > \frac{2^n}{n+1},$$

since $2^n/(n+1)$ is the average of the numbers in the row, and the largest number is certainly at least as large as the average.

These bounds already give a pretty good idea about the size of $\binom{n}{n/2}$. Take, say, $n = 500$. Then we get

$$\frac{2^{500}}{501} < \binom{500}{250} < 2^{500}.$$

If we want to know, say, the number of digits of $\binom{500}{250}$, we just have to take the logarithm (in base 10) of it. From the bounds above, we get

$$500 \lg 2 - \lg 501 < \lg \binom{500}{250} < 500 \lg 2.$$

Since $\lg 501 < 3$, this inequality gives the number of digits with a very small error: if we guess that it is $(500 \lg 2) - 1$, rounded up (which is 150) then we are off by at most 2 (actually, 150 is the true value).

Using the Stirling Formula, one can get a much better approximation of this largest entry.

$$\binom{n}{n/2} \sim \frac{2^n}{\sqrt{\pi n/2}}. \tag{9}$$

5.9 (a) Give a combinatorial argument to show that $2^n > \binom{n}{4}$.

(b) Use this inequality to show that $2^n > n^3$ if n is large enough.

5.10 Show how to use the Stirling Formula to derive (9).

So we know that the largest entry in the n-th row of the Pascal triangle is in the middle, and we know approximately how large this element is. We also know that going either left or right, the elements begin to drop. How fast do they drop? We show that if we move away from the middle of the row, quite soon the numbers drop to less than a half of the maximum. (We pick this 1/2 only for illustration; we shall see that one can estimate similarly how far away from the middle we have to move before the entries drop to a third, or to one percent, of the middle entry.) This quantifies our observation that the large binomial coefficients are concentrated in the middle of the row.

It is important to point out that in the arguments below, using calculus would give stronger results; we only give here as much as we can without appealing to calculus.

Again, we consider the case when n is even; then we can write $n = 2m$, where m is a positive integer. The middle entry is $\binom{2m}{m}$. Consider the binomial coefficient that is t steps before the middle: this is $\binom{2m}{m-t}$. We want to compare it with the largest coefficient. We shall choose the value of t later; this way our calculations will be valid for any value of t such that $0 \le t \le m$, and so they will be applicable in different situations.

We already know that $\binom{2m}{m} > \binom{2m}{m-t}$; we also know by (8) that going left from $\binom{2m}{m}$, the entries drop by factors $\frac{m}{m+1}$, then by $\frac{m-1}{m+2}$, then by $\frac{m-2}{m+3}$, etc. The last drop is by a factor of $\frac{m-t+1}{m+t}$.

Multiplying these factors together, we get

$$\binom{2m}{m-t} \Big/ \binom{2m}{m} = \frac{m(m-1)\dots(m-t+1)}{(m+1)(m+2)\dots(m+t)}.$$

So we have a formula for this ratio, but how do we know for which value does it become less than 1/3? We could write up the inequality

$$\frac{m(m-1)\dots(m-t+1)}{(m+1)(m+2)\dots(m+t)} < \frac{1}{2},$$

and solve it for t (as we did when we proved that the entries are increasing to the middle), but this would lead to inequalities that are too complicated to solve.

So we have to do some arithmetic trickery here. We start with looking at the reciprocal ratio:

$$\binom{2m}{m} \Big/ \binom{2m}{m-t} = \frac{(m+1)(m+2)\dots(m+t)}{m(m-1)\dots(m-t+1)}. \tag{10}$$

and write it in the following form:

$$\frac{(m+t)(m+t-1)\dots(m+1)}{m(m-1)\dots(m-t+1)} = \frac{m+t}{m} \cdot \frac{m+t-1}{m-1} \cdot \dots \cdot \frac{m+1}{m-t+1}$$
$$= \left(1+\frac{t}{m}\right) \cdot \left(1+\frac{t}{m-1}\right) \cdot \dots \cdot \left(1+\frac{t}{m-t+1}\right).$$

If we replace the numbers $m, m-1, \dots, m-t+1$ in the denominator by m, we decrease the value of each factor and thereby the value of the whole product:

$$\left(1+\frac{t}{m}\right) \cdot \left(1+\frac{t}{m-1}\right) \cdot \dots \cdot \left(1+\frac{t}{m-t+1}\right) \geq \left(1+\frac{t}{m}\right)^t. \tag{11}$$

It is still not easy to see how large this expression is; the base is close to 1, but the exponent may be large. We can get a simpler expression if we use the Binomial Theorem and then retain just the first two terms:

$$\left(1+\frac{t}{m}\right)^t = 1 + \binom{t}{1}\frac{t}{m} + \binom{t}{2}\left(\frac{t}{m}\right)^2 + \dots > 1 + \binom{t}{1}\frac{t}{m} = 1 + \frac{t^2}{m}.$$

To sum up, we conclude that

Theorem 5.1 *For all integers* $0 \leq t \leq m$,

$$\binom{2m}{m} \Big/ \binom{2m}{m-t} > 1 + \frac{t^2}{m}.$$

Let us choose t to be the least integer that is not smaller than $\sqrt{m}$ (in notation: $\lceil\sqrt{m}\rceil$). Then $t^2 \geq m$, and we get the following:

$$\binom{2m}{m-\lceil\sqrt{m}\rceil} < \frac{1}{2}\binom{2m}{m}.$$

5.11 Find a number c such that if $t > c\sqrt{m}$, then

$$\binom{2m}{m-t} < \frac{1}{100}\binom{2m}{m}.$$

Consider, for example, the 500^{th} row. Since $\sqrt{500} = 22.36\dots$, it follows that the entry $\binom{500}{227}$ is less than half of the largest entry $\binom{500}{250}$. This argument only gives an *upper bound* on how far we have to go; it does not say that the entries closer to the middle are all larger than a third of the middle entry; in fact, the entry $\binom{500}{236}$ is already smaller than a $\frac{1}{2}\binom{500}{250}$, but the entry $\binom{500}{237}$ is larger.

Next we prove that even if we sum all entries outside a narrow range in the middle, we get a small fraction of the sum of all entries. (This fact will be very important when we will apply these results in probability theory.)

We need an inequality that is a generalization of the inequality in Theorem 5.1. We state is as a "lemma"; this means an auxiliary result that may not be so interesting in itself, but will be important in proving some other theorem.

Lemma 5.1 *For all integers $0 \le t, s \le m$ such that $t + s \le m$,*

$$\binom{2m}{m-s} \Big/ \binom{2m}{m-t-s} > \frac{t^2}{m}.$$

For $s = 0$, this lemma says the same as Theorem 5.1. We leave the proof of it as an exercise.

5.12 (a) Prove Lemma 5.1, by following the proof of Theorem 5.1.

(b) Show that Lemma 5.1 follows from Theorem 5.1, if one observes that as s increases, the binomial coefficient $\binom{2m}{m-t-s}$ decreases faster than the binomial coefficient $\binom{2m}{m-s}$.

(c) Show that by doing the calculations more carefully, the lower bound of t^2/m in Lemma 5.1 can be improved to $1 + t(2t + s)/m$.

Now we state the theorem about the sum of the "small" binomial coefficients.

Theorem 5.2 *Let $0 \le k \le m$ and let $t = \lceil \sqrt{km} \rceil$. Then*

$$\binom{2m}{0} + \binom{2m}{1} + \ldots + \binom{2m}{m-t-1} < \frac{1}{2k} 2^{2m}. \tag{12}$$

To digest the meaning of this, choose $k = 100$. The quantity 2^{2m} on the right hand side is the sum of *all* binomial coefficient in row No. $2m$ of the Pascal Triangle; so the theorem says that if we take the sum of the first $m - t$ (where $t = \lceil 10\sqrt{m} \rceil$) then we get less than half percent of the total sum. By the symmetry of the Pascal Triangle, the sum of the last $m - t$ entries in this row will be the same, so the $2t + 1$ remaining terms in the middle make up 99 percent of the sum.

To prove this theorem, let us compare the sum on the left hand side of (12) with the sum

$$\binom{2m}{t} + \binom{2m}{t+1} + \ldots + \binom{2m}{m-1}. \tag{13}$$

Let us note right away that the sum in (13) is clearly less than $2^{2m}/2$, since even if we add the mirror image of this part of the row of the Pascal Triangle, we get less than 2^{2m}. Now for the comparison, we have

$$\binom{2m}{m-t-1} \le \frac{1}{k} \binom{2m}{m-1}$$

by Lemma 5.1 (check the computation!), and

$$\binom{2m}{m-t-2} < \frac{1}{k} \binom{2m}{m-2},$$

and similarly, each term on the left hand side of (13) is less that a hundredth of the corresponding term in (13). Hence we get that

$$\binom{2m}{0} + \binom{2m}{1} + \ldots + \binom{2m}{m-2t} < \frac{1}{k}\binom{2m}{t} + \binom{2m}{t+1} + \ldots + \binom{2m}{m-t} < \frac{1}{2k}2^{2m}.$$

This proves the theorem.

6 Combinatorial probability

6.1 Events and probabilities

Probability theory is one of the most important areas of mathematics from the point of view of applications. In this book, we do not attempt to introduce even the most basic notions of probability theory; our only goal is to illustrate the importance of combinatorial results about the Pascal Triangle by explaning a key result in probability theory, the Law of Large Numbers. To do so, we have to talk a little about what probability is.

If we make an observation about our world, or carry out an experiment, the outcome will always depend on chance (to a varying degree). Think of the weather, the stockmarket, or a medical experiment. Probability theory is a way of modeling this dependence on chance.

We start with making a mental list of all possible outcomes of the experiment (or observation, which we don't need to distringuish). These possible outcomes form a set S. Perhaps the simplest experiment is tossing a coin. This has two outcomes: H (head) and T (tail). So in this case $S = \{H, T\}$. As another example, the outcomes of throwing a dice form the set $S = \{1, 2, 3, 4, 5, 6\}$. In this book, we assume that the set $S = \{s_1, s_2, \ldots, s_k\}$ of possible outcomes of our experiment is finite. The set S is often called a *sample space*.

Every subset of S is called an *event* (the event that the observed outcome falls in this subset). So if we are throwing a dice, the subset $\{2, 4, 6\} \subseteq S$ can be thought of as the event that we have thrown an even number.

The intersection of two subsets corresponds to the event that both events occur; for example, the subset $L \cap E = \{4, 6\}$ corresponds to the event that we throw a better-than-average number that is also even. Two events A and B (i.e., two subsets of S) are called *exclusive* if the never occur at the same time, *i.e.*, $A \cap B = \emptyset$. For example, the event $O = \{1, 3, 5\}$ that the outcome of tossing a dice is odd and the event E that it is even are exclusive, since $E \cap O = \emptyset$.

6.1 What event does the union of two subsets corresponds to?

So let $S = \{s_1, s_2, \ldots, s_n\}$ be the set of possible outcomes of an experiment. To get a probability space we assume that each outcome $s_i \in S$ has a "probability" $\mathsf{P}(s_i)$ such that

(a) $\mathsf{P}(s_i) \geq 0$ for all $s_i \in S$,

and

(b) $\mathsf{P}(s_1) + \mathsf{P}(s_2) + \ldots + \mathsf{P}(s_k) = 1$.

Then we call S, together with these probabilities, a *probability space*. For example, if we toss a "fair" coin, the $\mathsf{P}(H) = \mathsf{P}(T) = 1/2$. If the dice in our example is of good quality, then we will have $\mathsf{P}(i) = 1/6$ for every outcome i.

A probability space in which every outcome has the same probability is called a *uniform probability space*. We shall only discuss uniform spaces here, since they are the easiest to imagine and they are the best for the illustration of combinatorial methods. But you should be warned that in more complicated modelling, non-uniform probability spaces are very often needed. For example, if we are observing if a day is rainy or not, we will have a 2-element sample space $S = \{\text{RAINY}, \text{NON-RAINY}\}$, but these two will typically *not* have the same probability.

The probability of an event $A \subseteq S$ is defined as the sum of probabilities of outcomes in A, and is denoted by $\mathsf{P}(A)$. If the probability space is uniform, then the probability of A is

$$\mathsf{P}(A) = \frac{|A|}{|S|} = \frac{|A|}{k}.$$

6.2 Prove that the probability of any event is at most 1.

6.3 What is the probability of the event E that we throw an even number with the dice? What is the probability of the event $T = \{3, 6\}$ that we toss a number that is divisible by 3?

6.4 Prove that if A and B are exclusive, then $\mathsf{P}(A) + \mathsf{P}(B) = \mathsf{P}(A \cap B)$.

6.5 Prove that for any two events A and B,

$$\mathsf{P}(A \cap B) + \mathsf{P}(A \cup B) = \mathsf{P}(A) + \mathsf{P}(B).$$

6.2 Independent repetition of an experiment

Let us repeat our experiment n times. We can consider this as a single big experiment, and a possible outcome of this repeated experiment is a sequence of length n, consisting of elements of S. Thus the sample space corresponding to this repeated experiment is the set S^n of such sequences. Thus the number of outcomes of this "big" experiment is k^n. We consider every sequence equally likely, which means that we consider it a uniform probability space. Thus if $(a_1, a_2, \ldots, a_n)$ is an outcome of the "big" experiment, then we have

$$\mathsf{P}(a_1, a_2, \ldots, a_n) = \frac{1}{k^n}.$$

As an example, consider the experiment of tossing a coin twice. Then $S = \{H, T\}$ (head,tail) for a single coin toss, and so the sample space for the two coin tosses is $\{HH, HT, TH, TT\}$. The probability of each of these outcomes is $1/4$.

This definition intends to model the situation where the outcome of each repeated experiment is independent of the previous outcomes, in the everyday sense that "there cannot possibly be any measurable influence of one experiment on the other". We cannot go here into the philosophical questions that this notion raises; all we can do is to give a mathematical definition that we can check on examples that it expresses the informal notion above.

A key notion in probability is that if *independence* of events. Informally, this means that information about one (whether or not it occured) does not influence the probability of the other. Formally, two events A and B are *independent* if $\mathsf{P}(A \cap B) = \mathsf{P}(A)\mathsf{P}(B)$. For example, if $E = \{2, 4, 6\}$ is the event that the result of throwing a dice is even, and $T = \{3, 6\}$ is the event that it is a multiple of 3, then E and the event T are independent: we have $E \cap T = \{6\}$ (the only possibility to throw a number that is even and divisible by 3 is to throw 6), and hence

$$\mathsf{P}(E \cap T) = \frac{1}{6} = \frac{1}{2} \cdot \frac{1}{3} = \mathsf{P}(E)\mathsf{P}(T).$$

Consider again the experiment of tossing a coin twice. Let A be the event that the first toss is head; let B be the event that the second toss is head. Then we have $\mathsf{P}(A) = \mathsf{P}(HH) + \mathsf{P}(HT) = 1/4 + 1/4 = 1/2$, similarly $\mathsf{P}(B) = 1/2$, and $\mathsf{P}(A \cap B) = \mathsf{P}(HH) = 1/4 = (1/2) \cdot (1/2)$. Thus A and B are independent events (as they should be).

6.6 Which pairs of the events E, O, T, L are independent? Which pairs are exclusive?

6.7 Show that $\emptyset$ is independent from every event. Is there any other event with this property?

6.8 Consider an experiment with sample space S repeated n times ($n \geq 2$). Let $s \in S$. Let A be the event that the first outcome is s, and let B be the event that the last outcome is s. Prove that A and B are independent.

6.3 The Law of Large Numbers

In this section we study an experiment that consists of n independent coin tosses. For simplicity, assume that n is even, so that $n = 2m$ for some integer m. Every outcome is a sequence of length n, in which each element is either H or T. A typical outcome would look like this:

$$HHTTTHTHTTHTHHHHTHTT$$

(for $n = 20$).

The "Law of Large Numbers" says that if we toss a coin many times, the number of 'heads' will be about the same as the number of 'tails'. How can we make this statement precise? Certainly, this will not *always* be true; one can be extremely lucky or unlucky, and have a winning or loosing streak of arbitrary length. Also, we can't claim that the number of heads is equal to the number of tails; only that they are close. The following theorem is the simplest form of the Law of Large Numbers.

Theorem 6.1 *Let $0 \leq t \leq m$. Then the probability that out of $2m$ independent coin tosses, the number of heads is less than $m - t$ or larger than $m + t$, is at most m/t^2.*

Let us discuss this a little. If $t < \sqrt{m}$ then the assertion does not say anything, since we know anyhow that the probability of any event is at most 1, and the upper bound given is larger than 1. But is we choose, say, $t = 10\sqrt{m}$, then we get an interesting special case:

Corollary 6.1 *With probability at least .99, the number of heads among $2m$ independent coin tosses is between $m - 10\sqrt{m}$ and $m + 10\sqrt{m}$.*

Of course, we need that $m > 10\sqrt{m}$, or $m > 100$, for this to have any meaning. But after all, we are talking about Large Numbers! If m is very large, then $10\sqrt{m}$ is much smaller than m, so we get that the number of heads is very close to m. For example, if $m = 1,000,000$ then $10\sqrt{m} = m/100$, and so it follows that with probability at least .99, the number of heads is within 1 percent of $m = n/2$.

We would like to show that the Law of Large Numbers is not a mysterious force, but a simple consequence of the properties of binomial coefficients.

Let A_k denote the event that we toss exactly k heads. It is clear that the events A_k are mutually exclusive. It is also clear that for every outcome of the experiment, exactly one of the A_k occurs.

The number of outcomes for which A_k occurs is the number of sequences of length n consisting of k 'heads' and $n - k$ 'tails'. If we specify which of the n positions are 'heads', we are done. This can be done in $\binom{n}{k}$ ways, so the set A_k has $\binom{n}{k}$ elements. Since the total number of outcomes is 2^n, we get the following.

$$\mathsf{P}(A_k) = \frac{\binom{n}{k}}{2^n}.$$

What is the probability that the number of heads is far from the expected, which is $m = n/2$? Say, it is less than $m - t$ or larger than $m + t$, where t is any positive integer not larger than m? Using exercise 6.1, we see that the probability that this happens is

$$\frac{1}{2^{2m}}\left(\binom{2m}{0} + \binom{2m}{1} + \ldots + \binom{2m}{m-t-1} + \binom{2m}{m+t+1} + \ldots + \binom{2m}{2m-1} + \binom{2m}{2m}\right).$$

Now we can use Theorem 5.2, with $k = t^2/m$, and get that

$$\binom{2m}{0} + \binom{2m}{1} + \ldots + \binom{2m}{m-t-1} < \frac{m}{2t^2}2^{2m}.$$

By the symmetry of the Pascal Triangle, we also have

$$\binom{2m}{m+t+1} + \ldots + \binom{2m}{2m-1} + \binom{2m}{2m} < \frac{m}{2t^2}2^{2m}.$$

Hence we get that the probability that we toss either less than $m-t$ or more than $m+t$ heads is less than m/t^2. This proves the theorem.

7 Fibonacci numbers

7.1 Fibonacci's exercise

In the 13^{th} century, the Italian mathematician Leonardo Fibonacci studied the following (not too realistic) exercise:

A farmer raises rabbits. Each rabbit gives birth to one rabbit when it turns 2 months old, and then to one rabbit each month. Rabbits never die, and we ignore hares. How many rabbits will the farmer have in the n-th month, if he starts with one newborn rabbit?

It is easy to figure out the answer for small values of n. He has 1 rabbit in the first month and 1 rabbit in the second month, since the rabbit has to be 2 months old before starting to reproduce. He has 2 rabbits during the third months, and 3 rabbits during the fourth, since his first rabbit delivered a new one after the second and one after the third. After 4 months, the second rabbit also delivers a new rabbit, so two new rabbits are added. This means that the farmer will have 5 rabbits during the fifth month.

It is easy to follow the multiplication of rabbits for any number of months, if we notice that the number of new rabbits added after n months is just the same as the number of rabbits who are at least 2 months old, i.e., who were already there after during the $(k-1)$-st month. In other words, if we denote by F_n the number of rabbits during the n-th month, then we have, for $n = 2, 3, 4, \ldots$,

$$F_{n+1} = F_n + F_{n-1}. \tag{14}$$

We also know that $F_1 = 1$, $F_2 = 1$, $F_3 = 2$, $F_4 = 3$, $F_5 = 5$. It is convenient to define $F_0 = 0$; then equation (14) will remain valid for $n = 1$ as well. Using the equation (14), we can easily determine any number of terms in this sequence of numbers:

$$0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597 \ldots$$

The numbers in this sequence are called *Fibonacci numbers.*

We see that equation (14), together with the special values $F_0 = 0$ and $F_1 = 1$, uniquely determines the Fibonacci numbers. Thus we can consider (14), together with $F_0 = 0$ and $F_1 = 1$, as the definition of these numbers. This may seem as a somewhat unusual definition: instead of telling what F_n is (say, by a formula), we just give a rule that computes each Fibonacci number from the two previous numbers, and specify the first two values. Such a definition is called a *recurrence.* It is quite similar in spirit to induction (except that it not a proof technique, but a definition method), and is sometimes also called *definition by induction.*

7.1 Why do we have to specify exactly two of the elements to begin with? Why not one or three?

Before trying to say more about these numbers, let us consider another counting problem:

A staircase has n steps. You walk up taking one or two at a time. How many ways can you go up?

For $n = 1$, there is only 1 way. For $n = 2$, you have 2 choices: take one twice or two once. For $n = 3$, you have 3 choices: three single steps, or one sigle followed by one double, or one double followed by one single.

Now stop and try to guess what the answer is in general! If you guessed that the number of ways to go up on a stair with n steps is n, you are wrong. The next case, $n = 4$, gives 5 possibilities $(1+1+1+1,\ 2+1+1,\ 1+2+1,\ 1+1+2,\ 2+2)$.

So instead of guessing, let's try the following strategy. We denote by G_n the answer, and try to figure out what G_{n+1} is, assuming we know the value of G_k for $k \le n$. If we start with a single step, we have G_n ways to go up the remaining n steps. If we start with a double step, we have G_{n-1} ways to go up the remaining $n-1$ steps. Now these are all the possibilities, and so

$$G_{n+1} = G_n + G_{n-1}.$$

This equation is the same as the equation we have used to compute the Fibonacci numbers F_n. Does this means that $F_n = G_n$? Of course not, as we see by looking at the beginning values: for example, $F_3 = 2$ but $G_3 = 3$. However, it is easy to observe that all that happens is that the G_n are shifted by one:

$$G_n = F_{n+1}.$$

This is valid for $n = 1, 2$, and then of course it is valid for every n since the sequences $F_2, F_3, F_4 \dots$ and $G_1, G_2, G_3, \dots$ are computed by the same rule from their first two elements.

7.2 We have n dollars to spend. Every day we either by a candy for 1 dollar, or an icecream for 2 dollars. In how many ways can we spend the money?

7.2 Lots of identities

There are many interesting relations valid for the Fibonacci numbers. For example, what is the sum of the first n Fibonacci numbers? We have

$$\begin{aligned}
0 &= 0,\\
0+1 &= 1,\\
0+1+1 &= 2,\\
0+1+1+2 &= 4,\\
0+1+1+2+3 &= 7,\\
0+1+1+2+3+5 &= 12,\\
0+1+1+2+3+5+8 &= 20,\\
0+1+1+2+3+5+8+13 &= 33.
\end{aligned}$$

Staring at these numbers for a while, it is not hard to recognize that adding 1 to the right hand sides we get Fibonacci numbers; in fact, we get Fibonacci numbers two steps after the last summand. In formula:

$$F_0 + F_1 + F_2 + \ldots + F_n = F_{n+2} - 1.$$

Of course, at this point this is only a *conjecture*, an unproved mathematical statement we believe to be true. To prove it, we use induction on n (since the Fibonacci numbers are defined by recurrence, induction is the natural and often only proof method at hand).

We have already checked the validity of the statement for $n = 0$ and 1. Suppose that we know that the identity holds for the sum of the first $n-1$ Fibonacci numbers. Consider the sum of the first n Fibonacci numbers:

$$F_0 + F_1 + \ldots + F_n = (F_0 + F_1 + \ldots + F_{n-1}) + F_n = (F_{n+1} - 1) + F_n,$$

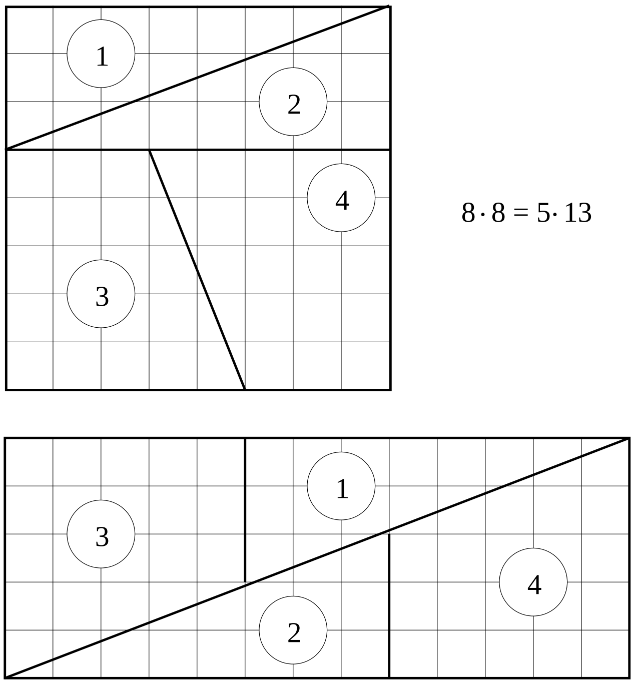

Figure 12: Proof of $65 = 64$

by the induction hypothesis. But now we can use the recurrence equation for the Fibonacci numbers, to get

$$(F_{n+1} - 1) + F_n = F_{n+2} - 1.$$

This completes the induction proof.

7.3 Prove that F_{3n} is even.

7.4 Prove that F_{5n} is divisible by 5.

7.5 Prove the following identities.

(a) $F_1 + F_3 + F_5 + \ldots + F_{2n-1} = F_{2n}$.

(b) $F_0 - F_1 + F_2 - F_3 + \ldots - F_{2n-1} + F_{2n} = F_{2n-1} - 1$.

(c) $F_0^2 + F_1^2 + F_2^2 + \ldots + F_n^2 = F_n \cdot F_{n+1}$.

(d) $F_{n-1}F_{n+1} - F_n^2 = (-1)^n$.

7.6 Mark the first entry (a 1) of any row of the Pascal triangle. Move one step East and one step Northeast, and mark the entry there. Repeat this until you get out of the triangle. Compute the sum of the entries you marked.

(a) What numbers do you get if you start from different rows? First "conjecture", than prove your answer.

(b) Formulate this fact as an identity involving binomial coefficients.

7.7 Cut a chessboard into 4 pieces as shown in Figure 12 and assemble a 5×13 rectangle from them. Does this prove that $5 \cdot 13 = 8^2$? Where are we cheating? What does this have to do with Fibonacci numbers?

7.3 A formula for the Fibonacci numbers

How large are the Fibonacci numbers? Is there a simple formula that expresses F_n as a function of n?

An easy way out, at least for the author of a book, is to state the answer right away:

Theorem 7.1 *The Fibonacci numbers are given by the formula*

$$F_n = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^n - (\left(\frac{1-\sqrt{5}}{2}\right)^n\right).$$

It is straightforward to check that this formula gives the right value for $n = 0, 1$, and then one can prove its validity for all n by induction.

7.8 Prove Theorem 7.1 by induction on n.

Do you feel cheated by this? You should; while it is of course logically correct what we did, one would like to see more: how can one arrive at such a formula? What should we try if we face a similar, but different recurrence?

So let us forget Theorem 7.1 for a while and let us try to find a formula for F_n "from scratch".

One thing we can try is to experiment. The Fibonacci numbers grow quite fast; how fast? Let's compute the ratio of consecutive Fibonacci numbers:

$$\frac{1}{1} = 1, \qquad \frac{2}{1} = 2, \qquad \frac{3}{2} = 1.5, \qquad \frac{5}{3} = 1.666666667, \qquad \frac{8}{5} = 1.600000000,$$

$$\frac{13}{8} = 1.625000000, \qquad \frac{21}{13} = 1.615384615, \qquad \frac{34}{21} = 1.619047619, \qquad \frac{55}{34} = 1.617647059,$$

$$\frac{89}{55} = 1.618181818, \qquad \frac{144}{89} = 1.617977528, \qquad \frac{233}{144} = 1.618055556, \qquad \frac{377}{233} = 1.618025751.$$

It seems that the ratio of consecutive Fibonacci numbers is very close to 1.618, at least if we ignore the first few values. This suggests that the Fibonacci numbers behave like a geometric progression. So let's see if there is any geometric progression that satisfies the same recurrence as the Fibonacci numbers. Let $G_n = c \cdot q^n$ be a geometric progression ($c, q \neq 0$). Then

$$G_{n+1} = G_n + G_{n-1}$$

translates into

$$c \cdot q^{n+1} = c \cdot q^n + c \cdot q^{n-1},$$

which after simplification becomes

$$q^2 = q + 1.$$

(So the number c disappears: what this means that if we find a sequence of this form that satisfies Fibonacci's recurrence, then we can change c in it to any other real number, and get another sequence that satisfies the recurrence.)

What we have is a quadratic equation for q, which we can solve and get

$$q_1 = \frac{1+\sqrt{5}}{2} \approx 1.618034, \qquad q_2 = \frac{1-\sqrt{5}}{2} \approx -0.618034.$$

So we have found two kinds of geometric progressions that satisfy the same recurrence as the Fibonacci numbers:

$$G_n = c\left(\frac{1+\sqrt{5}}{2}\right)^n, \qquad G'_n = c\left(\frac{1-\sqrt{5}}{2}\right)^n$$

(where c is an arbitrary constant). Unfortunately, neither G_n nor G'_n gives the Fibonacci sequence: for one, $G_0 = G'_0 = c$ while $F_0 = 0$. But notice that the sequence $G_n - G'_n$ also satisfies the recurrence:

$$G_{n+1} - G'_{n+1} = (G_n + G_{n-1}) - (G'_n - G'_{n-1} = (G_n - G'_n) + (G_{n-1} - G'_{n-1})$$

(using that G_n and G'_n satisfy the recurrence). Now we have matched the first value F_0, since $G_0 - G'_0 = 0$. What about the next one? We have $G_1 - G'_1 = c\sqrt{5}$. We can match this with $F_1 = 1$ if we choose $c = 1/\sqrt{5}$.

Now we have two sequences: F_n and $G_n - G'_n$, that both begin with the same two numbers, and satisfy the same recurrence. Hence they must be the same: $F_n = G_n - G'_n$.

Now you can substitute for the values of G_n and G'_n, and see that we got the formula in the Theorem!

Let us include a little discussion of the formula we just derived. The first base in the exponential expression is $q_1 = (1 + \sqrt{5}/2) \approx 1.618034 > 1$, while the second base q_2 is between -1 and 0. Hence if n increases, then G_n will become very large, while $|G'_n| < 1/2$ and in fact G'_n becomes very small. This means that

$$F_n \approx G_n = \frac{1}{\sqrt{5}}\left(1+\sqrt{5}\right)^n,$$

where the term we ignore is less than 1/2 (and tends t0 0 if n tends to infinity); this means that F_n is the integer nearest to G_n.

The base $(1+\sqrt{5})/2$ is called the *golden ratio*, and it comes up all over mathematics; for example, it is the ratio between the diagonal and side of a regular pentagon.

7.9 Define a sequence of integers H_n by $H_0 = 1$, $H_1 = 3$, and $H_{n+1} = H_n + H_{n-1}$. Show that H_n can be expressed in the form $a \cdot q_1^n + b \cdot q_2^n$ (where q_1 and q_2 are the same numbers as in the proof above), and find the values of a and b.

7.10 Define a sequence of integers I_n by $I_0 = 0$, $I_1 = 1$, and $I_{n+1} = 4I_n + I_{n-1}$. Find a formula for I_n.

8 Integers, divisors, and primes

In this chapter we discuss properties of integers. This area of mathematics is called *number theory*, and it is a truely venerable field: its roots go back about 2500 years, to the very beginning of Greek mathematics. One maight think that after 2500 years of research, one would know essentially everything about the subject. But we shall see that this is not the case: there are very simple, natural questions which we cannot answer; and there are other simple, natural questions to which an answer has only been found in the last years!

8.1 Divisibility of integers

We start with some very basic notions concerning integers. Let a and b be two integers. We say that *a divides b*, or *a is a divisor of b*, or *b is a multiple of a* (these phrases mean the same thing), if there exists an integer m such that $b = am$. In notation: $a|b$. If a is not a divisor of b, then we write $a \not| b$. If $a \neq 0$, then this means that the ratio b/a is an integer.

If $a \not| b$, and $a > 0$, then we can still divide b by a with remainder. The remainder r of the division $b : a$ is an integer that satisfies $0 \leq r < a$. If the quotient of the division with remainder is q, then we have

$$b = aq + r.$$

This is a very useful way of thinking about a division with remainder.

You have probably seen these notions before; the following exercises should help you check if you remember enough.

8.1 Check (using the definition) that $1|a$, $-1|a$, $a|a$ and $-a|a$ for every integer a.

8.2 What does it mean for a, in more everyday terms, if (a) $2|a$; (b) $2 \not| a$; (c) $0|a$.

8.3 (a) Prove that

(a) if $a|b$ and $b|c$ then $a|c$;

(b) if $a|b$ and $a|c$ then $a|b+c$ and $a|b-c$;

(c) if $a, b > 0$ and $a|b$ then $a \leq b$;

(d) if $a|b$ and $b|a$ then either $a = b$ or $a = -b$.

8.4 Let r be the remainder of the division $b : a$. Assume that $c|a$ and $c|b$. Prove that $c|r$.

8.5 Assume that $a|b$, and $a, b > 0$. Let r be the remainder of the division $c : a$, and let s be the remainder of the division $c : b$. What is the remainder of the division $s : a$?

8.6 (a) Prove that for every integer a, $a - 1|a^2 - 1$.

(b) More generally, for every integer a and positive integer m,

$$a - 1|a^n - 1.$$

8.2 Primes and their history

An integer $p > 1$ is called a *prime* if it is not divisible by any integer other than $1, -1, p$ and $-p$. Another way of saying this is that an integer $p > 1$ is a prime if it cannot be written as the product of two smaller positive integers. An integer $n > 1$ that is not a prime is called *composite* (the number 1 is considered neither prime, nor composite). Thus $2, 3, 5, 7, 11$ are

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101,
103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197,
199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311,
313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431,
433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541, 547, 557,
563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641, 643, 647, 653, 659, 661,
673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773, 787, 797, 809,
811, 821, 823, 827, 829, 839, 853, 857, 859, 863, 877, 881, 883, 887, 907, 911, 919, 929, 937,
941, 947, 953, 967, 971, 977, 983, 991, 997, 1009, 1013, 1019, 1021, 1031, 1033, 1039, 1049,
1051, 1061, 1063, 1069, 1087, 1091, 1093, 1097, 1103, 1109, 1117, 1123, 1129, 1151, 1153,
1163, 1171, 1181, 1187, 1193, 1201, 1213, 1217, 1223, 1229, 1231, 1237, 1249, 1259, 1277,
1279, 1283, 1289, 1291, 1297, 1301, 1303, 1307, 1319, 1321, 1327, 1361, 1367, 1373, 1381,
1399, 1409, 1423, 1427, 1429, 1433, 1439, 1447, 1451, 1453, 1459, 1471, 1481, 1483, 1487,
1489, 1493, 1499, 1511, 1523, 1531, 1543, 1549, 1553, 1559, 1567, 1571, 1579, 1583, 1597,
1601, 1607, 1609, 1613, 1619, 1621, 1627, 1637, 1657, 1663, 1667, 1669, 1693, 1697, 1699,
1709, 1721, 1723, 1733, 1741, 1747, 1753, 1759, 1777, 1783, 1787, 1789, 1801, 1811, 1823,
1831, 1847, 1861, 1867, 1871, 1873, 1877, 1879, 1889, 1901, 1907, 1913, 1931, 1933, 1949,
1951, 1973, 1979, 1987, 1993, 1997, 1999, 2003, 2011, 2017, 2027, 2029, 2039, 2053, 2063,
2069, 2081, 2083, 2087, 2089, 2099, 2111, 2113, 2129, 2131, 2137, 2141, 2143, 2153, 2161,
2179, 2203, 2207, 2213, 2221, 2237, 2239, 2243, 2251, 2267, 2269, 2273, 2281, 2287, 2293,
2297, 2309, 2311, 2333, 2339, 2341, 2347, 2351, 2357, 2371, 2377, 2381, 2383, 2389, 2393,
2399, 2411, 2417, 2423, 2437, 2441, 2447, 2459, 2467, 2473, 2477, 2503, 2521, 2531, 2539,
2543, 2549, 2551, 2557, 2579, 2591, 2593, 2609, 2617, 2621, 2633, 2647, 2657, 2659, 2663,
2671, 2677, 2683, 2687, 2689, 2693, 2699, 2707, 2711, 2713, 2719, 2729, 2731, 2741, 2749,
2753, 2767, 2777, 2789, 2791, 2797, 2801, 2803, 2819, 2833, 2837, 2843, 2851, 2857, 2861,
2879, 2887, 2897, 2903, 2909, 2917, 2927, 2939, 2953, 2957, 2963, 2969, 2971, 2999, 3001,
3011, 3019, 3023, 3037, 3041, 3049, 3061, 3067, 3079, 3083, 3089, 3109, 3119, 3121, 3137,
3163, 3167, 3169, 3181, 3187, 3191, 3203, 3209, 3217, 3221, 3229, 3251, 3253, 3257, 3259,
3271, 3299, 3301, 3307, 3313, 3319, 3323, 3329, 3331, 3343, 3347, 3359, 3361, 3371, 3373,
3389, 3391, 3407, 3413, 3433, 3449, 3457, 3461, 3463, 3467, 3469, 3491, 3499, 3511, 3517,
3527, 3529, 3533, 3539, 3541, 3547, 3557, 3559, 3571

Table 1: The first 500 primes

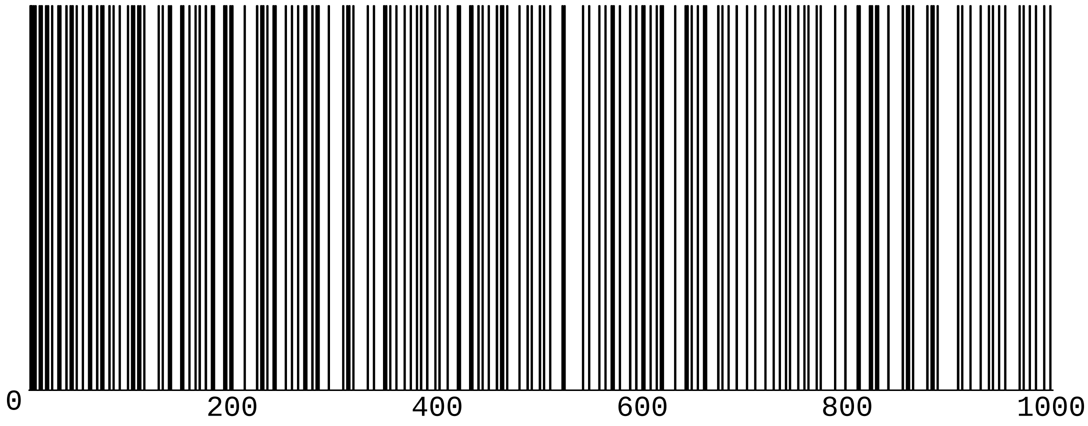

Figure 13: A bar chart of primes up to 1000

primes, but $4 = 2 \cdot 2$, $6 = 2 \cdot 3$, $8 = 2 \cdot 4$, $9 = 3 \cdot 3$, $10 = 2 \cdot 5$ are not primes. Table 1 contains the first 500 primes.

Primes have fascinated people ever since ancient times. Their sequence seems very irregular, yet on closer inspection it seems to carry a lot of hidden structure. The ancient Greeks already knew that there are infinitely many such numbers. (Not only did they know this; they proved it!)

It was not easy to learn more about primes. Their sequence has holes and dense spots (see also figure 13). How large are these holes? For example, is there a prime number with any given number of digits? The answer to this question will be important for us when we discuss cryptography. The answer to this question is in the affirmative, but this fact was not proved until the last century, and many similar questions are open even today.

A new wave of developments in the theory of prime numbers came with the spread of computers. How do you decide about a positive integer n whether it is a prime? Surely this is a finite problem (you can try out all smaller positive integers to see if any of them is a proper divisor), but such simple methods fail as soon as the number of digits is more than 10 or so.

It is only in the last 20 years since much more efficient algorithms (computer programs) exist to test if a given integer is a prime. We will get a glimpse of these methods later. Using these methods, one can now rather easily determine about a number with 1000 digits whether it is a prime or not.

If an integer larger than 1 is not a prime itself, then it can be written as a product of primes: we can write it as a product of two smaller positive integers; if one of these is not a prime, we write it as the product of two smaller integers etc; sooner or later we must end up with only primes. The ancient Greeks also knew (and proved!) a subtler fact about this representation: that *it is unique*. What this means is that there is no other way of writing n as a product of primes (except, of course, we can multiply the same primes in a different order). To prove this takes some sophistication, and to recognize the necessity of such a result was quite an accomplishment; but this is all more than 2000 years old!

It is really surprising than that, even today, no efficient way is known to *find* such a decomposition. Of course, powerful supercomputers and massively parallel systems can be used to find decompositions by brute force for fairly large numbers; the current record is around a 140 digits, and the difficulty grows very fast (exponentially) with number of digits. To find the prime decomposition of a number with 400 digits, by any of the known methods, is way beyond the possibilities of computers in the foreseeable future.

8.3 Factorization into primes

We have seen that every integer larger than 1 that is not a prime itself can be written as a product of primes. We can even say that *every* positive integer which can be written as a product of primes: primes can be considered as "products with one factor", and the integer 1 can be thought of as the "empty product". With this in mind, we can state and prove the following theorem, annonced above, sometimes called the "Fundamental Theorem of Number Theory".

Theorem 8.1 *Every positive integer can be written as the product of primes, and this factorization is unique up to the order of the prime factors.*

We prove this theorem by a version of induction, which is sometimes called the "minimal criminal" argument. The proof is indirect: we suppose that the assertion is false, and using this assumption, we derive a logical contradiction.

So assume that there exists an integer with two different factorizations; call such an integer a "criminal". There may be many criminals, and we consider the smallest one:

$$n = p_1 \cdot p_2 \cdot \ldots \cdot p_m = q_1 \cdot q_2 \cdot \ldots \cdot q_k \,.$$

We may assume that p_1 is the smallest prime occuring in these factorizations. (Indeed, if necessary, we can interchange the left hand side and the right hand side so that the smallest prime in any of the two factorizations occurs on the left hand side; and then change the order of the factors on the left hand side so that the smallest factor comes first. In the usual slang of mathematics, we say that we may assume that p_1 is the smallest prime *without loss of generality.*)

The number p_1 cannot occur among the factors q_i, otherwise we can divide both sides by p_1 and get a smaller criminal.

Divide each q_i by p_1 with residue: $q_i = p_i a_i + r_i$, where $0 \le r_i < p_1$. We know that $r_i \neq 0$, since a prime cannot be a divisor of another prime.

Let $n' = r_1 \cdot \ldots \cdot r_k$. We show that n' is a smaller criminal. Trivially $n' = r_1 r_1 \ldots r_k < q_1 q_2 \ldots q_k = n$. We show that n' too has two different factorizations into primes. One of these can be obtained from the definition $n' = r_1 r_2 \ldots r_k$. Here the factors may not be primes, but we can break them down into products of primes, so that we end up with a decomposition of n'.

To get another decomposition, we observe that $p_1 | n'$. Indeed, we can write the definition of n' in the form

$$n' = (q_1 - a_1 p_1)(q_2 - a_2 p_1) \ldots (q_k - a_k p_1),$$

and if we expand, then every term will be divisible by p_1. Now we divide n' by p_1 and then continue to factor n'/p_1, to get a factorization of n'.

But are these two factorizations different? Yes! The prime p_1 occurs in the second, but it cannot occur in the first, where every prime factor is smaller than p_1.

Thus we have found a smaller criminal. Since n was supposed to be the smallest among all criminals, this is a contradiction. The only way to resolve this contradiction is to conclude the our "indirect assumption" was false, and no integer can have two different prime factorizations.

As an application of Theorem 8.1, we prove a fact that was known to the Pythagoreans (students of Pythagoras) in the 6th century B.C.

Theorem 8.2 *The number $\sqrt{2}$ is irrational.*

We give an indirect proof again: we suppose that $\sqrt{2}$ is rational, and derive a contradiction. What the indirect assumption means is that $\sqrt{2}$ can be written as the quotient of two positive integers: $\sqrt{2} = \frac{a}{b}$. Squaring both sides and rearranging, we get $2b^2 = a^2$.

Now consider the prime factorization of both sides, and in particular, the prime number 2 on both sides. Suppose that 2 occurs m times in the prime factorization of a and n times in the prime factorization of b. Then it occurs $2m$ times in the prime factorization of a^2 and thus it occurs $2m+1$ times in the prime factorization of $2a^2$. On the other hand, it occurs $2n$ times in the prime factorization of b^2. Since $2a^2 = b^2$ and the prime factorization is unique, we must have $2m+1 = 2n$. But this is impossible since $2m+1$ is odd but $2n$ is even. This contradiction proves that $\sqrt{2}$ must be irrational.

8.7 Are there any even primes?

8.8 (a) Prove that if p is a prime, a and b are integers, and $p|ab$, then either $p|a$ or $p|b$ (or both).

(b) Suppose that a and b are integers and $a|b$. Also suppose that p is a prime and $p|b$ but $p \not| a$. Prove that p is a divisor of the ratio b/a.

8.9 Prove that the prime factorization of a number n contains at most $\log_2 n$ factors.

8.10 Let p be a prime and $1 \le a \le p-1$. Consider the numbers $a, 2a, 3a, \dots, (p-1)a$. Divide each of them by p, to get residues $r_1, r_2, \dots, r_{p-1}$. Prove that every integer from 1 to $p-1$ occurs *exactly once* among these residues.

[Hint: First prove that no residue can occur twice.]

8.11 Prove that if p is a prime, then $\sqrt{p}$ is irrational. More generally, prove that if n is an integer that is not a square, then $\sqrt{n}$ is irrational.

8.12 Try to formulate and prove an even more general theorem about the irrationality of the numbers $\sqrt[k]{n}$.

8.4 On the set of primes

The following theorem was also known to Euclid.

Theorem 8.3 *There are infinitely many primes.*

What we need to do is to show that for every positive integer n, there is a prime number larger than n. To this end, consider the number $n!+1$, and any prime divisor p of it. We show that $p > n$. Again, we use an indirect proof, supposing that $p \le n$ and deriving a contradiction. If $p \le n$ then $p|n!$, since it is one of the integers whose product is $n!$. We also know that $p|n!+1$, and so p is a divisor of the difference $(n!+1)-n!=1$. But this is impossible, and thus p must be larger than n.

If we look at various charts or tables of primes, our main impression is that there is a lot of irregularity in them. For example, figure 13 represents each prime up to 200 by a bar. We see large "gaps" and then we also see primes that are very close. We can prove that these gaps get larger and larger as we consider larger and larger numbers; somewhere out there is a string of 100 consecutive composite numbers, somewhere (much farther away) there is a string of a 1000, etc. To state this in a mathematical form:

Theorem 8.4 *For every positive integer k, there exist k consecutive composite integers.*

We can prove this theorem by an argument quite similar to the proof of theorem 8.3. Let $n=k+1$ and consider the numbers

$$n!+2,\ n!+3,\ \ldots,\ n!+n.$$

Can any of these be a prime? The answer is no: the first number is even, since $n!$ and 2 are both even. The second number is divisible by 3, since $n!$ and 3 are both divisible by 3 (assuming that $n>2$). In general $n!+i$ is divisible by i, for every $i=2,3,\ldots,n$. Hence these numbers cannot be primes, and so we have found $n-1=k$ consecutive composite numbers.

What about the opposite question, finding primes very close to each other? Since all primes except 2 are odd, the difference of two primes must be at least two, except for 2 and 3. Two primes whose difference is 2 are called *twin primes.* Thus $(3,5)$, $(5,7)$, $(11,13)$, $(17,19)$ are twin primes. Looking at the table of the first 500 primes, we find many twin primes; extensive computation shows that there are twin primes with hundreds of digits. However, it is not known whether there are infinitely many twin primes! (Almost certainly there are, but no proof of this fact has been found, in spite of the efforts of many mathematicians for over 2000 years!)

Another way of turning Theorem 8.4 around: how large can be these gaps? Could it happen that there is no prime at all with, say, 100 digits? This is again a very difficult question, but here we do know the answer. (No, this does not happen.)

One of the most important questions about primes is: how many primes are there up to a given number n? We denote the number of primes up to n by $\pi(n)$. Figure 14 illustrates the graph of this function in the range of 1 to 100, and Figure 15, in the range of 1 to 2000. We can see that the function grows reasonably smoothly, and that its slope decreases slowly. An exact formula for $\pi(n)$ is certainly impossible to obtain. Around the turn of the century, a powerful result called the Prime Number Theorem was proved by two French mathematicians, Hadamard and de la Vallée-Poussain. From this result it follows that we can find prime numbers with a prescribed number of digits, and up to a third of the digits also prescribed. For example, there exist prime numbers with 30 digits starting with 1234567890.

Theorem 8.5 (The Prime Number Theorem) *Let $\pi(n)$ denote the number of primes among $1,2,\ldots,n$. Then*

$$\pi(n) \sim \frac{n}{\ln n}$$

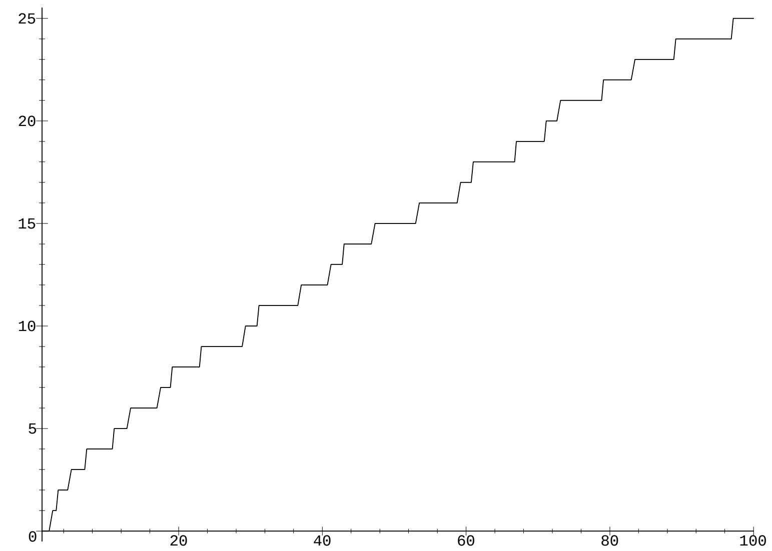

Figure 14: The graph of $\pi(n)$ from 1 to 100

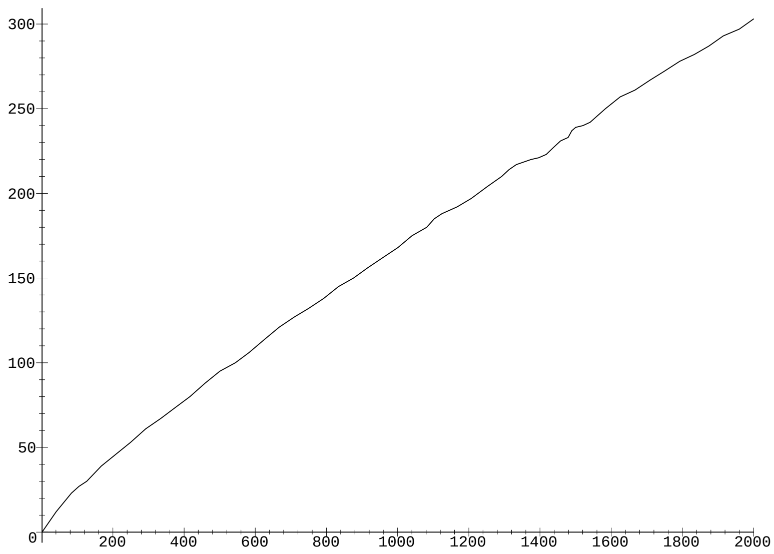

Figure 15: The graph of $\pi(n)$ from 1 to 2000

(Here $\ln n$ means the "natural logarithm", i.e., logarith to the base $e = 2.718281\ldots$. Also recall that the notation means that

$$\frac{\pi(n)}{\frac{n}{\ln n}}$$

will be arbitrarily close to 1 if n is sufficiently large.)

The proof of the prime number theorem is very difficult; the fact that the number of primes up to n is about $n/\ln n$ was observed emperically in the 18^{th} century, but it took more than 100 years until Hadamard and de la Vallée-Poussin proved it in 1896.

As an illustration of the use of this theorem, let us find the answer to a question that we have posed in the introduction: how many primes with k digits are there? We get the answer by subtracting the number of primes up to 10^{k-1} from the number of primes up to 10^k. By the Prime Number Theorem, this number is about

$$\frac{10^k}{k\ln 10} - \frac{10^{k-1}}{(k-1)\ln 10} = \frac{(9k-10)10^{k-1}}{k(k-1)\ln 10}$$

Since

$$\frac{9k-10}{k-1} = 9 - \frac{1}{k-1}$$

is very close to 9 if k is large, we get that the number of primes with k digits is approximately

$$\frac{9\cdot 10^{k-1}}{k\ln 10}.$$

Comparing this with the total number of positive integers with k digits, which we know is $10^k - 10^{k-1} = 9\cdot 10^{k-1}$, we get

$$\frac{9\cdot 10^{k-1}}{k\ln 10\cdot 9\cdot 10^{k-1})} = \frac{1}{(\ln 10)k} \approx \frac{1}{2.3k}.$$

Thus among the integers with k digits, one in $2.3k$ is a prime.

(Warning: This argument is not precise; the main source of concern is that the in the prime number theorem, we only stated the $\pi(n)$ is close to $n/\ln n$ if n is sufficiently large. One can say more about how large n has to be to have, say, an error less than 1 percent, but this leads to even more difficult questions, which are even today not completely resolved.)

There are many other simple observations one can make by looking at tables of primes, but they tend to be very difficult and most of them are not resolved even today, in some cases after 2,500 years of attempts. We have already mentioned the problem whether there are infinitely many twin primes.

Another famous unsolved problem is *Goldbach's conjecture.* This states that every even integer larger than 2 can be written as the sum of two primes. (Goldbach also formulated a conjecture about odd numbers: every odd integer larger than 5 can be written as the sum of three primes. This conjecture was essentially proved, using very deep methods, by Vinogradov in the 1930's. We said "essentially" since the proof only works for numbers that are very large, and the possibility of a finite number of exceptions remains open.)

Suppose that we have an integer n and want to know how soon after n can we be sure to find a prime. For example, how small, or large, is the first prime with at least 100 digits? Our proof of the infinity of primes gives that for every n, there is a prime between n and

$n! + 1$. This is a very week statement; it say that, say, there is a prime between 10 and $10! + 1 = 3628801$, while of course the next prime is 11. Chebychev proved in the last century that there is always a prime between n and $2n$. It is now proved that there is always a prime between two consecutive cubes (say, between $27 = 3^3$ and $64 = 4^3$). It is another old and famous unsolved problem whether there is always a prime between two consecutive squares. (Try this out: you'll in fact find many primes. For example, between $100 = 10^2$ and $121 = 11^2$ we find $101, 103, 107, 109, 113$.)

8.5 Fermat's "Little" Theorem

Primes are important because we can compose every integer from them; but it turns out that they also have many surprising properties. One of these was discovered by the French mathematician Pierre Fermat (1601–1655). Before stating it, we prove a simple lemma.

Lemma 8.1 *If p is a prime and $1 < k < p$, then $p|\binom{p}{k}$.*

Proof. We know by theorem 4.2 that

$$\binom{p}{k} = \frac{p(p-1)\cdot\ldots\cdot(p-k+1)}{k(k-1)\cdot\ldots\cdot 1}.$$

Now here p divides the numerator but not the denominator (since all factors in the denominator are smaller than p, and we know (exercise 8.3(a) that if a prime p does not divide any of these factors, then it does not divide the product. Hence it follows (see exercise 8.3(b)) that p is a divisor of $\binom{p}{k}$. $\square$

Now we state the famous "Little" Theorem of Fermat.

Theorem 8.6 *If p is a prime and a is an integer, then $p|a^p - a$.*

Proof. We prove this theorem by induction on a. The assertion is trivially true if $a = 0$. Let $a > 0$, and write $a = b + 1$. Then

$$\begin{aligned} a^p - a &= (b+1)^p - (b+1) = b^p + \binom{p}{1}b^{p-1} + \ldots + \binom{p}{p-1}b + 1 - b - 1 \\ &= (b^p - b) + \binom{p}{1}b^{p-1} + \ldots + \binom{p}{p-1}b. \end{aligned}$$

Here the expression $b^p - b$ in paranthesis is divisible by p by the induction hypothesis, while the other terms are divisible by p by lemma 8.1. It follows that $a^p - a$ is also divisible by p, which completes the induction. $\square$

Let us make a remark about the history of mathematics here. Fermat is most famous for his "Last" theorem, which is the following assertion:

if $n > 2$, then the sum of the n-th powers of two positive integers is never an n-th power of a positive integer.

(The assumption that $n > 2$ is essential: there are nmany examples of two squares whose sum is a third square: for example, $3^2 + 4^2 = 5^2$, or $5^2 + 12^2 = 13^2$.)

Fermat claimed in a note that he proved this, but never wrote down the proof. This statement remained perhaps the most famous unsolved problem in mathematics until 1995, when Andrew Wiles finally proved it.

8.13 Show by examples that neither the assertion in lemma 8.1 nor Fermat's Little Theorem remain valid if we drop the assumption that p is a prime.

8.14 Consider a regular p-gon, and all k-subsets of the set of its vertices, where $1 \leq k \leq p1$. Put all these k-subsets into a number of boxes: we put two k-subsets into the same box if they can be rotated into each other. For example, all k-subsets consisting of k consecutive vertices will belong to one and the same box.

(a) Prove that if p is a prime, then each box will contain exactly p of these sets.

(b) Show by an example that (a) does not remain true if we drop the assumption that p is a prime.

(c) Use (a) to give a new proof of Lemma 8.1.

8.15 Imagine numbers written in base a, with at most p digits. Put two numbers in the same box if they arise by cyclic shift from each other. How many will be in each class? Give a new proof of Fermat's theorem this way.

8.16 Give a third proof of Fermat's "Little Theorem" based on exercise 8.3.
[Hint: consider the product $a(2a)(3a)\dots((p-1)a)$.]

8.6 The Euclidean Algorithm

So far, we have discussed several notions and results concerning integers. Now we turn our attention to the question of how to do computations in connection with these results. How to decide whether or not a given number is a prime? How to find the prime factorization of a number?

We can do basic arithmetic: addition, subtraction, multiplication, division with remainder efficiently, and will not discuss this here.

The key to a more advanced algorithmic number theory is an algorithm that computes the *greatest common divisor* of two positive integers a and b. This is defined as the largest positive integer that is a divisor of both of them. (Since 1 is always a common divisor, and no common divisor is larger than either integer, this definition makes sense.) We say that two integers are *relatively prime* if their greatest common divisor is 1. The greatest common divisor of a and b is denoted by $\gcd(a, b)$. Thus

$$\gcd(1,6) = 1, \qquad \gcd(2,6) = 2, \qquad \gcd(3,6) = 3, \qquad \gcd(4,6) = 2,$$
$$\gcd(5,6) = 1, \qquad \gcd(6,6) = 6.$$

Two numbers whose greatest common divisor is 1 are called *relatively prime*. It will be convenient to also define $\gcd(a, 0) = a$ for every $a \geq 0$.

A somewhat similar notion is the *least common multiple* of two integers, which is the least positive integer that is a multiple of both integers, and denote by $\mathrm{lcm}(a, b)$. For example,

$$\mathrm{lcm}(1,6) = 6, \qquad \mathrm{lcm}(2,6) = 6, \qquad \mathrm{lcm}(3,6) = 6, \qquad \mathrm{lcm}(4,6) = 12,$$
$$\mathrm{lcm}(5,6) = 30, \qquad \mathrm{lcm}(6,6) = 6$$

The greatest common divisor of two positive integers can be found quite simply by using their prime factorizations: look at the common prime factors, raise them to the smaller of the two exponents, and take the product of these prime powers. For example, $300 = 2^2 \cdot 3 \cdot 5^2$ and $18 = 2 \cdot 3^3$, and hence $\gcd(300, 18) = 2 \cdot 3 = 6$.

The trouble with this method is that it is very difficult to find the prime factorization of large integers. The algorithm to be discussed in this section will compute the greatest common divisor of two integers in a much faster way, without finding their prime factorization. This algorithm is an important ingredient of almost all other algorithms involving computation with integers. (And, as we see it from its name, it goes back to the great Greek mathematicians!)

8.17 Show that if a and b are positive integers with $a|b$, then $\gcd(a,b) = a$.

8.18 (a) $\gcd(a,b) = \gcd(a, b-a)$.

(b) Let r be the remainder if we divide b by a. Then $\gcd(a,b) = \gcd(a,r)$.

8.19 (a) If a is even and b is odd, then $\gcd(a,b) = \gcd(a/2, b)$.

(b) If both a and b are even, then $\gcd(a,b) = 2\gcd(a/2, b/2)$.

8.20 How can you express the least common multiple of two integers, if you know the prime factorization of each?

8.21 Suppose that you are given two integers, and you know the prime factorization of one of them. Describe a way of computing the greatest common divisor of these numbers.

8.22 Prove that for any two integers a and b,

$$\gcd(a,b)\mathrm{lcm}(a,b) = ab.$$

Now we turn to Euclid's Algorithm. Let a and b be two positive integers, and suppose that $a < b$. The algorithm is based on two simple facts, already familiar as exercises 8.6 and 8.6.

Suppose that we are given two positive integers a and b, and we want to find their g.c.d. Here is what we do:

1. If $a > b$ then we interchange a and b.
2. If $a > 0$, divide b by a, to get a remainder r. Replace b by r and return to 1.
3. Else (if $a = 0$), return b as the g.c.d. and halt.

When you carry out the algorithm, especially by hand, there is no reason to interchange a and b if $a < b$: we can simply divide the larger number by the smaller (with remainder), and replace the larger number by the remainder if the remainder is not 0. Let us do some examples.

$$\begin{aligned}
\gcd(300,18) &= \gcd(12,18) = \gcd(12,6) = 6. \\
\gcd(101,100) &= \gcd(1,100) = 1. \\
\gcd(89,55) &= \gcd(34,55) = \gcd(34,21) = \gcd(13,21) = \gcd(13,8) = \gcd(5,8) \\
&= \gcd(5,3) = \gcd(2,3) = \gcd(2,1) = 1.
\end{aligned}$$

You can check in each case (using a prime factorization of the numbers) that the result is indeed the g.c.d.

If we describe an algorithm, the first thing to worry about is whether it terminates at all. So why is the euclidean algorithm finite? This is easy: the numbers never increase, and one of them decreases any time step 2 is executed, so it cannot last infinitely long.

Then of course we have to make sure that our algorithm yields what we need. This is clear: step 1 (interchanging the numbers) trivially does not change the g.c.d., step 3 (replacing the larger number by the remainder of a division) does not change the g.c.d. by exercise 8.6(b). And when we halt at step 2, the number returned is indeed the g.c.d. of the two current numbers by exercise 8.6.

A third, and more subtle, question you should ask when designing an algorithm: how long does it take? How many steps will it make before it terminates? We can get a bound from the argument that proves finite termination: since one or the other number decreases any time the 1-2 loop is executed, it will certainly halt in less than $a+b$ iterations. This is really not a great time bound: if we apply the euclidean algorithm to two numbers with 100 digits, then it says that it will not last longer than $2 \cdot 10^{100}$ steps, which is astronomical and therefore useless. But this is only a most pessimistic bound; the examples seem to show that the algorithm terminates much faster than this.

But the examples also suggest that question is quite delicate. We see that the euclidean algorithm may be quite different in length, depending on the numbers in question. Some of the possible observations made from this examples are contained in the following exercises.

> **8.23** Show that the euclidean algorithm can terminate in two steps for arbitrarily large positive integers, even if the g.c.d is 1.
>
> **8.24** Describe the euclidean algorithm applied to two consecutive Fibonacci numbers. Use your description to show that the euclidean algorithm can last arbitrarily many steps.

So what *can* we say about how long does the euclidean algorithm last? The key to the answer is the following lemma:

Lemma 8.2 *During the execution of the euclidean algorithm, the product of the two current numbers drops by a factor of two in each iteration.*

To see that this is so, consider the step when the pair (a,b) $(a<b)$ is replaced by the pair (r,a), where r is the remainder of b when divided by a. Then we have $r<a$ and hence $b \geq a+r > 2r$. Thus $ar < \frac{1}{2}ab$ as claimed.

Suppose that we apply the euclidean algorithm to two numbers a and b and we make k steps. It follows by Lemma 8.2 that after the k steps, the product of the two current numbers will be at most $ab/2^k$. Since this is at least 1, we get that

$$ab \geq 2^k,$$

and hence

$$k \leq \log_2(ab) = \log_2 a + \log_2 b.$$

Thus we have proved the following.

Theorem 8.7 *The number of steps of the euclidean algorithm, applied to two positive integers a and b, is at most $\log_2 a + \log_2 b$.*

We have replaced the sum of the numbers by the sum of the logarithms of the numbers in the bound on the number of steps, which is a really substantial improvement. For example, the number of iterations in computing the g.c.d. of two 300-digit integers is less than $2\log_2 10^{300} = 600\log_2 10 < 2000$. Quite a bit less than our first estimate of 10^{100}. Note that $\log_2 a$ is less than the number of bits of a (when written in base 2), so we can say that the euclidean algorithm does not take more iterations than the number of bits needed to write down the numbers in base 2.

The theorem above gives only an upper bound on the number of steps the euclidean algorithm takes; we can be much more lucky: for example, when we apply the euclidean algorithm to two consecutive integers, it takes only one step. But sometimes, one cannot do much better. If you did exercise 8.6, you saw that when applied to two consecutive Fibonacci numbers F_k and F_{k+1}, the euclidean algorithm takes $k-1$ steps. On the other hand, the lemma above gives the bound

$$\log_2 F_k + \log_2 F_{k+1} \approx \log_2 \left(\frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^k \right) \log_2 \left(\frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^{k+1} \right)$$

$$= -\log_2 5 + (2k+1) \log_2 \left(\frac{1+\sqrt{5}}{2} \right) \approx 1.388k - 1.628,$$

so we have overestimated the number of steps only by a factor of about 1.388.

Fibonacci numbers are not only good to give examples of large numbers for which we can see how the euclidean algorith works; they are also useful in obtaining an even better bound on the number of steps. We state the result as an exercise. Its contents is that, in a sense, the euclidean algorithm is longest on two consecutive Fibonacci numbers.

8.25 Suppose that $a < b$ and the euclidean algorithm applied to a and b takes k steps. Prove that $a \geq F_k$ and $b \geq F_{k+1}$.

8.26 Consider the following version of the euclidean algorithm to compute $\gcd(a, b)$: (1) swap the numbers if necessary to have $a \leq b$; (2) if $a = 0$, then return b; (3) if $a \neq 0$, then replace b by $b - a$ and go to (1).

(a) Carry out this algorithm to compute $\gcd(19, 2)$.

(b) Show that the modified euclidean algorithm always terminates with the right answer.

(c) How long does this algorithm take, in the worst case, when applied to two 100-digit integers?

8.27 Consider the following version of the euclidean algorithm to compute $\gcd(a, b)$. Start with computing the largest power of 2 dividing both a and b. If this is 2^r, then divide a and b by 2^r. After this "preprocessing", do the following:

(1) Swap the numbers if necessary to have $a \leq b$.

(2) If $a \neq 0$, then check the parities of a and b; if a is even, and b is odd, then replace a by $a/2$; if both a and b are odd, then replace b by $b - a$; in each case, go to (1).

(3) if $a = 0$, then return $2^r b$ as the g.c.d.

Now come the exercises:

(a) Carry out this algorithm to compute $\gcd(19, 2)$.

(b) It seems that in step (2), we ignored the case when both a and b are even. Show that this never occurs.

(c) Show that the modified euclidean algorithm always terminates with the right answer.

(d) Show that this algorithm, when applied to two 100-digit integers, does not take more than 1500 iterations.

8.7 Testing for primality

Returning to a question proposed in the introduction of this chapter: how do you decide about a positive integer n whether it is a prime? You can try to see if n can be divided, without remainder, by any of the smaller integers (other than 1). This is, however, a very clumsy and inefficient procedure! To determine this way whether a number with 20 digits is a prime would take astronomical time.

You can do a little better by noticing that it is enough to try to divide n by the positive integers less than $\sqrt{n}$ (since if n can be written as the product of two integers larger than 1, then one of these integers will necessarily be less than $\sqrt{n}$). But this method is still hopelessly slow if the number of digits goes above 20.

A much more promising approach is offered by Fermat's "Little" Theorem. Its simplest nontrivial case says that *if p is a prime, then $p|2^p - 2$.* If we assume that p is odd (which only excludes the case $p = 2$), then we also know that $p|2^{p-1} - 1$.

What happens if we check the divisibility relation $n|2^{n-1} - 1$ for composite numbers? It obviously fails if n is even (no even number is a divisor of an odd number), so let's restrict our attention to add numbers. Here are some results:

$$9 \not| 2^8 - 1 = 255, \qquad 15 \not| 2^{14} - 1 = 16,383, \qquad 21 \not| 2^{20} - 1 = 1,048,575,$$

$$25 \not| 2^{24} - 1 = 16,777,215.$$

This suggests that perhaps we could test whether or not the number n is a prime by checking whether or not the relation $n|2^{n-1} - 1$ holds. This is a nice idea, but it has several major shortcomings.

First, it is easy to write up the formula $2^{n-1} - 1$, but it is quite a different matter to compute it! It seems that to get 2^{n-1}, we have to multiply $n - 2$ times by 2. For a 100 digit number, this is about 10^{100} steps, which we will never be able to carry out.

But, we can be tricky when we compute 2^{n-1}. Let us illustrate this on the example of 2^{24}: we could start with $2^3 = 8$, square it to get $2^6 = 62$, square it again to get $2^{12} = 4096$, and square it once more to get $2^{24} = 16,777,216$. Instead of 23 multiplications, we only needed 5.

It seems that this trick only worked because 24 was divisible by such a large power of 2, and we could compute 2^{24} by repeated squaring, starting from a small number. So let us show how to do a similar trick if the exponent is a less friendly integer. Let us compute, say, 2^{29}:

$$2^2 = 4, 2^3 = 8, 2^6 = 64, 2^7 = 128, 2^{14} = 16,384, 2^{28} = 268,435,456, 2^{29} = 536,870,912.$$

It is perhaps best to read this sequence backwards: if we have to compute an odd power of 2, we obtain it by multiplying the previous power by 2; if we have to compute an even power, we obtain it by squaring the appropriate smaller power.

8.28 Show that if n has k bits in base 2, then 2^n can be computed using less than $2k$ mupltiplications.

Thus we have shown how to overcome the first difficulty; but the computations we did above reveal the second: the numbers grow too large! Let's say that n has 100 digits; then 2^{n-1} is not only astronomical: the number of its digits is astronomical. We could never write it down, let alone check whether it is divisible by n.

The way out is to divide by n as soon as we get any number that is larger than n, and just work with the remainder of the division. For example, if we want to check whether $25|2^{24} - 1$, then we have to compute 2^{24}. As above, we start with computing $2^3 = 8$, then square it to get $2^6 = 64$. We immediately replace it by the remainder of the division $64 : 25$, which is 14. Then we compute 2^{12} by squaring 2^6, but instead we square 14 to get 196, which we replace by the remainder of the division $196 : 25$, which is 21. Finally, we obtain 2^{24} by squaring 2^{12}, but instead we square 21 to get 441, and then divide this by 25 to get the remainder 16. Since $16 - 1 = 15$ is not divisible by 25, it follows that 25 is not a prime.

This does not sound like an impressive conclusion, considering the triviality of the result, but this was only an illustration. If n has k bits in base 2, then as we have seen, it takes only $2k$ multiplications to compute 2^n, and all we have to do is one division (with remainder) in each step to keep the numbers small. We never have to deal with numbers larger than n^2. If n has 100 digits, then n^2 has 199 or 200 — not much fun to multiply by hand, but quite easily manageable by computers.

But here comes the third shortcoming of the primality test based on Fermat's Theorem. Suppose that we carry out the test for a number n. If it fails (that is, n is not a divisor of $2^{n-1} - 1$), then of course we know that n is not a prime. But suppose we find that $n | 2^{n-1} - 1$. Can we conclude that n is a prime? Fermat's Theorem certainly does not justify this conclusion. Are there composite numbers n for which $n | 2^{n-1} - 1$? Unfortunately, the answer is yes. The smallest such number is $341 = 11 \cdot 31$.

Here is an argument showing that 341 is a pseudoprime, without using a computer. We start with noticing that 11 divides $2^5 + 1 = 33$, and that 31 divides $2^5 - 1 = 31$. Therefore $341 = 11 \cdot 31$ divides $(2^5 + 1)(2^5 - 1) = 2^{10} - 1$. To finish, we invoke the result of exercise 8.1: it implies that $2^{10} - 1$ is a divisor of $(2^{10})^{34} - 1 = 2^{340} - 1$.

Such numbers are sometimes called pseudoprimes (fake primes). While such numbers are quite rare (there are only 22 of them between 1 and 10,000), they do show that our primality test can give a "false positive", and thus (in a strict mathematical sense) it is not a primality test at all.

Nevertheless, efficient primality testing is based on Fermat's Theorem. Below we sketch how it is done; the mathematical details are not as difficult as in the case of, say, the Prime Number Theorem, but they do go beyond the scope of this book.

One idea that comes to rescue is that we haven't used the full force of Fermat's Theorem: we can also check that $n | 3^n - 3$, $n | 5^n - 5$, etc. These can be carried out using the same tricks as described above. And in fact already the first of these rules out the "fake prime" 341. Unfortunately, some pseudoprimes are worse than others; for example, the number $561 = 3 \cdot 11 \cdot 17$ has the property that

$$561 | a^{561} - a$$

for *every* integer a. These pseudoprimes are called Carmicheal numbers, and it was believed that their existence kills every primality test based on Fermat's theorem.

But in the late 70's, M. Rabin and G. Miller found a very simple way to strengthen Fermat Theorem just a little bit, and thereby overcome the difficulty caused by Carmicheal numbers. We illustrate the method on the example of 561. We use some high school math, namely the identity $x^2 - 1 = (x - 1)(x + 1)$, to factor the number $a^{561} - 1$:

$$\begin{aligned}
a^{561} - a &= a(a^{560} - 1) = a(a^{280} - 1)(a^{280} + 1) && (15)\\
&= a(a^{140} - 1)(a^{140} + 1)(a^{280} + 1) && (16)\\
&= a(a^{140} - 1)(a^{140} + 1)(a^{280} + 1) && (17)\\
&= a(a^{70} - 1)(a^{70} + 1)(a^{140} + 1)(a^{280} + 1) && (18)
\end{aligned}$$

$$= a(a^{35}-1)(a^{35}+1)(a^{70}+1)(a^{140}+1)(a^{280}+1) \qquad (19)$$

Now suppose that 561 were a prime. Then by Fermat's Theorem, it must divide $a^{561} - a$, whatever a is (this in fact happens). Now if a prime divides a product, it divides one of the factors (exercise 8.3), and hence at least one of the relations

$$561|a \qquad 561|a^{35}-1 \qquad 561|a^{35}+1 \qquad 561|a^{70}+1 \qquad 561|a^{140}+1 \qquad 561|a^{280}+1$$

must hold. But already for $a = 2$, none of these relations hold.

The Miller-Rabin test is an elaboration of this idea. Given an odd integer $n > 1$ that we want to test for primality. We choose an integer a from the range $0 \le a \le n-1$ at random, and consider $a^n - a$. We factor it as $a(a^{n-1}-1)$, and then go on to factor it, using the identity $x^2 - 1 = (x-1)(x+1)$, as long as we can. Then we test that one of the factors must be divisible by n.

If the test fails, we can be sure that n is not a prime. But what happens if it succeeds? Unfortunately, this can still happen even if n is comnposite; but the crucial point is that *this test gives a false positive with probability less than* $1/2$ (remember we chose a random a).

Reaching a wrong conclusion half of the time does not sound good enough; but we can repeat the experiment several times. If we repeat it 10 times, the probability of a false positive is less than $2^{-10} < 1/1000$ (since to conclude that n is prime, all 10 runs must give a false positive, independently of each other). If we repeat the experiment 100 times, the probability of a false positive drops below $2^{-100} < 10^{-30}$, which is astronomically small.

Thus this test, when repeated sufficiently often, tests primality with error probability that is much less than the probability of, say, hardware failure, and therefore it is quite adequate for practical purposes. It is widely used in programs like Maple or Mathematica and in cryptographic systems.

Suppose that we test a number n for primality and find that it is composite. Then we would like to find its prime factorization. It is easy to see that instead of this, we could ask for less: for a decomposition of n into the product of two smaller positive integers: $n = ab$. If we have a method to find such a decomposition efficiently, then we can go on and test a and b for primality. If they are primes, we have found the prime factorization of n; if (say) a is not a prime, we can use our method to find a decomposition of a into the product of two smaller integers etc. Since n has at most $\log_2 n$ prime factors (exercise 8.3), we have to repeat this at most $\log_2 n$ times (which is less than the number of its bits).

But, unfortunately (or fortunately? see the next chapter on cryptography) no efficient method is known to write a composite number as a product of two smaller integers. It would be very important to find an efficient factorization method, or to give a mathematical proof that no such method exists; but we don't know what the answer is.

8.29 Show that $561|a^{561} - a$, for every integer a. [Hint: it suffices to prove that $3|a^{561} - a$, $11|a^{561} - 1$ and $17|a^{561} - a$. Prove these relations separately, using the method used in the proof of the fact that $341|2^{340} - 1$.]

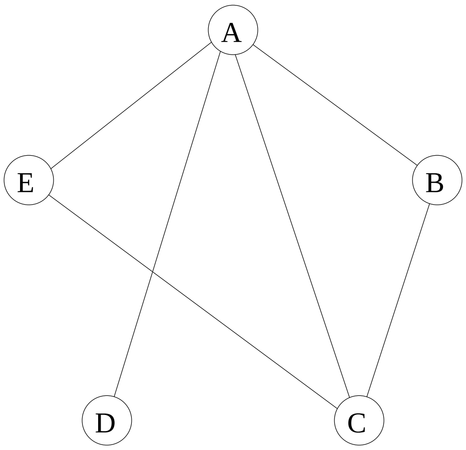

Figure 16: The graph depicting aquaitence between our friends

9 Graphs

9.1 Even and odd degrees

We start with the following exercise (admittedly of no practical significance).

Prove that at a party with 51 people, there is always a person who knows an even number of others. (We assume that acquaintance is mutual. There may be people who don't know each other. There may even be people who don't know anybody else — of course, such people know an even number of others, so the assertion is true if there is such a person.)

If you don't have any idea how to begin a solution, you should try to experiment. But how to experiment with such a problem? Should we find 51 names for the participants, then create, for each person, a list of those people he knows? This would be very tedious, and we would be lost among the data. It would be good to experiment with smaller numbers. But which number can we take instead of 51? It is easy to see that 50, for example, would not do: if, say, we have 50 people all know each other, then everybody knows 49 others, so there is no person with an even number of acquaintances. For the same reason, we could not replace 51 by 48, or 30, or any even number. Let's hope that this is all; let's try to prove that

at a party with an odd number of people, there is always a person who knows an even number of others.

Now we can at least experiment with smaller numbers. Let us have, say, 5 people: Alice, Bob, Carl, Diane and Eve. When they first met, Alice new everybody else, Bob and Carl new each other, and Carl also new Eve. So the numbers of acquaintances are : Alice 4, Bob 2, Carl 3, Diane 1 and Eve 2. So we have not only one but three people with an even number of acquaintances.

It is still rather tedious to consider examples by listing people and listing pairs knowing each other, and it is quite easy to make mistakes. We can, however, find a graphic illustration that helps a lot. We represent each person by a point in the plane (well, by a small circle, to make the picture nicer), and we connect two of these points by a segment if the people know each other. This simple drawing contains all the information we need (Figure 16).

A picture of this kind is called a *graph*. More exactly, a graph consists of a set of *nodes* (or *points*, or *vertices*, all these names are in use), and some pairs of these (not necessarily all

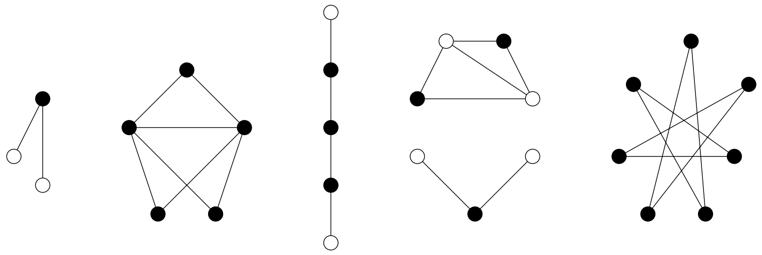

Figure 17: Some graphs with an odd number of nodes, with nodes of even degree marked

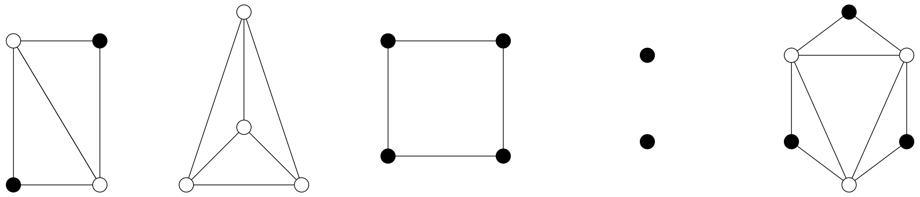

Figure 18: Some graphs with an even number of nodes, with nodes of even degree marked

pairs) are connected by *edges* (or *lines*). It does not matter whether these edges are straight of curvy; all that is important is which pair of nodes they connect. The set of nodes of a graph G is usually denoted by V; the set of edges, by E. Thus we write $G = (V, E)$ to indicate that the graph G has node set V and edge set E.

The only thing that matters about an edge is the pair of nodes it connects; hence the edges can be considered as 2-element subsets of V. This means that the edge connecting nodes u and v is just the set $\{u, v\}$. We'll further simplify notation and denote this edge by uv.

Coming back to our problem, we see that we can represent the party by a graph very conveniently. Our concern is the number of people known by a given person. We can read this off the graph by counting the number of edges leaving a given node. This number is called the *degree* (or *valency*) of the node. The degree of node v is denoted by $d(v)$. So A has degree 4, B has degree 2, etc. If Frank now arrives, and he does not know anybody, then we add a new node that is not connected to any other node. So this new node has degree 0.

In the language of graph theory, we want to prove:

if a graph has an odd number of nodes than it has a node with even degree.

Since it is easy to experiment with graphs, we can draw a lot of graphs with an odd number of nodes, and count the number of nodes with even degree (Fig. 17). We find that they contain $1, 5, 3, 3, 7$ such nodes. So we observe that not only is there always such a node, but the number of such nodes is odd.

Now this is a case when it is easier to prove more: if we formulate the following stronger statement:

if a graph has an odd number of nodes then the number of nodes with even degree is odd,

then we made an important step towards the solution! (Why is this statement stronger? Because 0 is not an odd number.) Let's try to find an even stronger statement by looking also at graphs with an even number of nodes. Experimenting on several small graphs again (Fig 18), we find that the number of nodes with even degree is $2, 0, 4, 2, 0$. So we conjecture that the following is true:

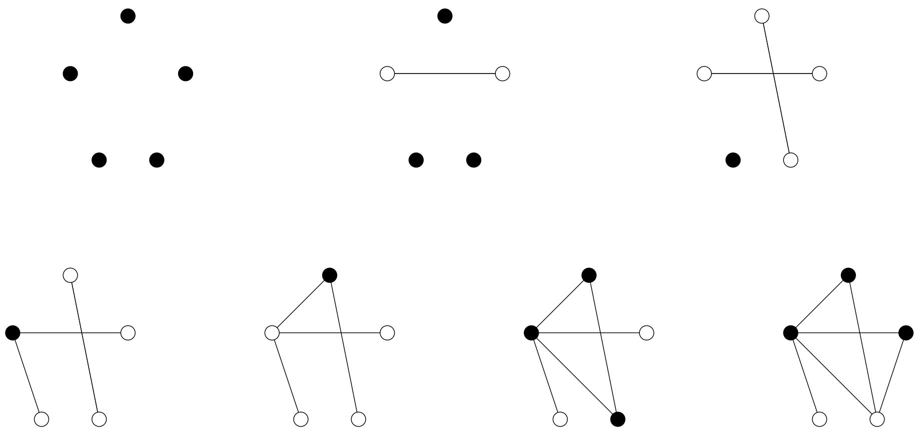

Figure 19: Building up a graph one edge at a time

if a graph has an even number of nodes then the number of nodes with even degree is even.

This is nicely parallel to the statement about graphs with an odd number of nodes, but it would be better to have a single common statement for the odd and even case. We get such a version if we look at the number of nodes with *odd*, rather than *even*, degree. This number is obtained by subtracting the number of nodes with even degree from the total number of nodes, and hence both statements will be implied by the following:

Theorem 9.1 *In every graph, the number of nodes with odd degree is even.*

We still have to prove this theorem. It seems that having made the statement stronger and more general in several steps, we made our task harder and harder. But in fact we got closer to the solution.

One way of proving the theorem is to build up the graph one edge at a time, and observe how the parities of the degrees change. An example is shown in Figure 19. We start with a graph with no edge, in which every degree is 0, and so the number of nodes with odd degree is 0, which is an even number.

Now if we connect two nodes by a new edge, we change the parity of the degrees at these nodes. In particular,

— if both endpoints of the new edge had even degree, we increase the number of nodes with odd degree by 2;

— if both endpoints of the new edge had odd degree, we decrease the number of nodes with odd degree by 2;

— if one endpoint of the new edge had even degree and the other had odd degree, then we don't change the number of nodes with odd degree.

Thus if the number of nodes with odd degree was even before adding the new edge, it remained even after this step. This proves the theorem. (Note that this is a proof by induction on the number of edges.)

Graphs are very handy in representing a large variety of situations, not only parties. It is quite natural to consider the graph whose nodes are towns and whose edges are highways (or railroads, or telephone lines) between these towns. We can use a graph to describe an electrical network, say the printed circuit on a card in your computer.

In fact, graphs can be used any situation where a "relation" between certain objects is defined. Graphs are used to describe bonds between atoms in a molecule; connections between cells in the brain; descendence between species, etc. Sometimes the nodes represent more abstract things: for example, they may represent stages of a large construction project, and an edge between two stages means that one arises from the other in a single phase of work. Or the nodes can represent all possible positions in a game (say, chess, although you don't really want to draw this graph), where we connect two nodes by an edge if one can be obtained from the other in a single move.

9.1 Find all graphs with $2, 3$ and 4 nodes.

9.2 (a) Is there a graph on 6 nodes with degrees $2, 3, 3, 3, 3, 3$?

(b) Is there a graph on 6 nodes with degrees $0, 1, 2, 3, 4, 5$?

(c) How many graphs are there on 4 nodes with degrees $1, 1, 2, 2$?

(d) How many graphs are there on 10 nodes with degrees $1, 1, 1, 1, 1, 1, 1, 1, 1, 1$?

9.3 At the end of the party, everybody knows everybody else. Draw the graph representing this situation. How many edges does it have?

9.4 Draw the graphs representing the bonds between atoms in (a) a water molecule; (b) a methane molecule; (c) two water molecules.

9.5 (a) Draw a graph with nodes representing the numbers $1, 2, \ldots, 10$, in which two nodes are connected by an edge if and only if one is a divisor of the other.

(b) Draw a graph with nodes representing the numbers $1, 2, \ldots, 10$, in which two nodes are connected by an edge if and only if they have no common divisor larger than 1.

(c) Find the number of edges and the degrees in these graphs, and check that theorem 9.1 holds.

9.6 What is the largest number of edges a graph with 10 nodes can have?

9.7 How many graphs are there on 20 nodes? (To make this question precise, we have to make sure we know what it means that two graphs are the same. For the purposes of this exercise, we consider the nodes given, and labeled say, as Alice, Bob, ... The graph consisting of a single edge connecting Alice and Bob is different from the graph consisting of a single edge connecting Eve and Frank.)

9.8 Formulate the following assertion as a theorem about graphs, and prove it: At every party one can find two people who know the same number of other people (like Bob and Eve in our first example).

It will be instructive to give another proof of the theorem formulated in the last section. This will hinge on the answer to the following question: How many edges does a graph have? This can be answered easily, if we think back to the problem of counting handshakes: for each node, we count the edges that leave that node (this is the degree of the node). If we sum these numbers, we count every edge twice. So dividing the sum by two, we get the number of edges. Let us formulate this observation as a theorem:

Theorem 9.2 *The sum of degrees of all nodes in a graph is twice the number of edges.*

In particular, we see that the sum of degrees in any graph is an even number. If we omit the even terms from this sum, we still get an even number. So the sum of odd degrees is even. But this is only possible if the number of odd degrees is even (since the sum of an odd number of odd numbers is odd). Thus we have obtained a new proof of Theorem 9.1.

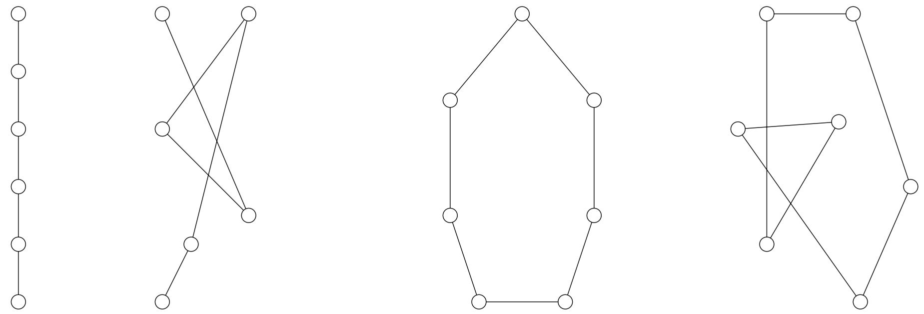

Figure 20: Two paths and two cycles

9.2 Paths, cycles, and connectivity

Let us get acquainted with some special kinds of graphs. The simplest graphs are the *empty graphs*, having any number of nodes but no edges.

We get another very simple kind of graphs if we take n nodes and connect any two of them by an edge. Such a graph is called a *complete graph* (or a *clique*). A complete graph with n nodes has $\binom{n}{2}$ edges (recall exercise 9.1).

Let us draw n nodes in a row and connect the consecutive ones by an edge. This way we obtain a graph with $n-1$ edges, which is called a *path*. The first and last nodes in the row are called the *endpoints* of the path. If we also connect the last node to the first, we obtain a *cycle* (or *circuit*). Of course, we can draw the same graph in many other ways, placing the nodes elsewhere, and we may get edges that intersect (Figure 20).

A graph H is called a *subgraph* of a graph G if it can be obtained from G by deleting some of its edges and nodes (of course, if we delete a node we automatically delete all the edges that connect it to other nodes).

9.9 Find all complete graphs, paths and cycles among the graphs in the previous figures.

9.10 How many subgraphs does an empty graph on n nodes have? How many subgraphs does a triangle have?

A key notion in graph theory is that of a *connected* graph. It is heuristically clear what this should mean, but it is also easy to formulate the property as follows: a graph G is connected if every two nodes of the graph can be connected by a path in G. To be more precise: a graph G is connected if for every two nodes u and v, there exists a path with endpoints u and v that is a subgraph of G (Figure 21).

It will be useful to include a little discussion of this notion. Suppose that nodes a and b can be connected by a path P in our graph. Also suppose that nodes b and c can be connected by a path Q. Can a and c be connected by a path? The answer seems to be obviously "yes", since we can just go from a to b and then from b to c. But there is a difficulty: concatenating (joining together) the two paths may not yield a path from a to c, since P and Q may intersect each other (Figure 22). But we can construct a path from a to c easily: Let us follow the path P to its first common node d with Q; then let us follow Q to c. Then the nodes we traversed are all distinct. Indeed, the nodes on the first part of our walk are distinct because they are nodes of the path P; similarly, the nodes on the second part are distinct because they are nodes of the path Q; finally, any node of the first part must be distinct from any node of the second part (except, of course, the node d), because d is the *first* common node of the two

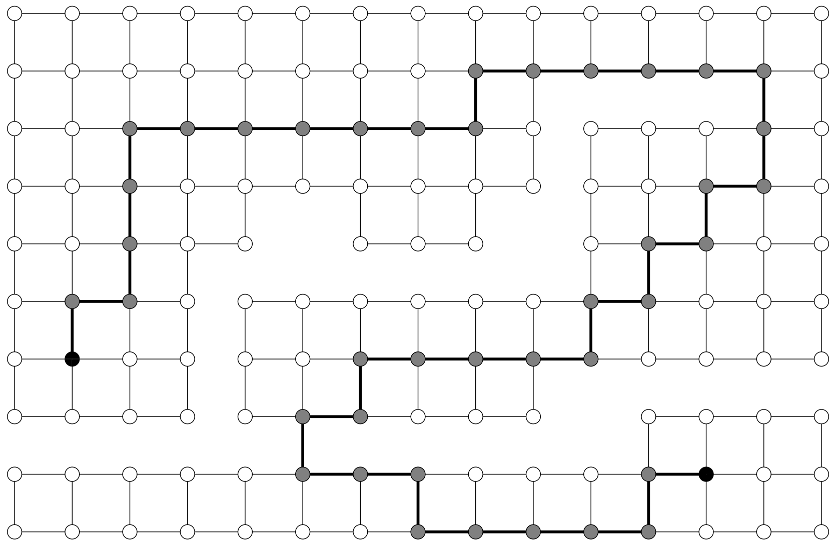

Figure 21: A path in a graph connecting two nodes

paths and so the nodes of P that we passed through before d are not nodes of Q at all. Hence the nodes and edges we have traversed form a path from a to c as claimed.[3]

9.11 Is the proof as given above valid if (a) the node a lies on the path Q; (b) the paths P and Q have no node in common except b.

9.12 (a) We delete an edge e from a connected graph G. Show by an example that the remaining graph may not be connected.

(b) Prove that if we assume that the deleted edge e belongs to a cycle that is a subgraph of G, then the remaining graph is connected.

9.13 Let G be a graph and let u and v be two nodes of G. A *walk* in G from u to v is a sequence of nodes $(w_0, w_1, \dots, w_k)$ such that $w_0 = u$, $w_k = v$, and consecutive nodes are connected by an edge, i.e., $w_i w_{i+1} \in E$ for $i = 0, 1, \dots, k-1$.

(a) Prove that if there is a walk in G from u to v, then G contains a path connecting u to v.

(b) Use part (a) to give another proof of the fact that if G contains a path connecting a and b, and also a path connecting b to c, then it contains a path connecting a to c.

9.14 Let G be a graph, and let $H_1 = (V_1, E_1)$ and $H_2 = (V_2, E_2)$ be two subgraphs of G that are connected. Assume that H_1 and H_2 have at least one node in common. Form their union, i.e., the subgraph $H = (V', E')$, where $V' = V_1 \cup V_2$ and $E' = E_1 \cup E_2$. Prove that H is connected.

Let G be a graph that is not necessarily connected. G will have connected subgraphs; for example, the subgraph consisting of a single node (and no edge) is connected. A *connected component* H is a maximal subgraph that is connected; in other words, H is a connected

[3]We have given the more details of this proof than was perhaps necessary. One should note, however, that when arguing about paths and cycles in graphs, it is easy to draw pictures (on paper or mentally) that make implicit assumptions and are, therefore, misleading. For example, when joining together two paths, one's first mental image is a single (longer) path, which may not be the case.

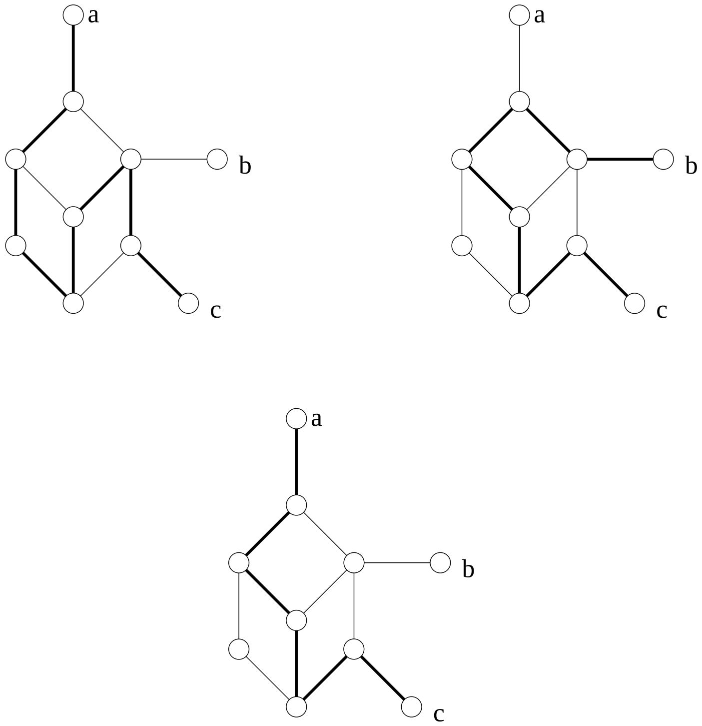

Figure 22: How to select a path from a to c, given a path from a to b and a path from b to c?

component if it is connected but every other subgraph of G that contains H is disconnected. It is clear that every node of G belongs to some connected component. It follows by exercise 9.2 that different connected components of G have no node in common (else, their union would be a connected subgraph containing both of them). In other words, every node of G is contained in a unique connected component.

9.15 Determine the connected components of the graphs constructed in exercise 9.1.

9.16 Prove that no edge of G can connect nodes in different connected components.

9.17 Prove that a node v is a node of the connected component of G containing node u if and only if g contains a path connecting u to v.

9.18 Prove that a graph with n nodes and more than $\binom{n-1}{2}$ edges is always connected.

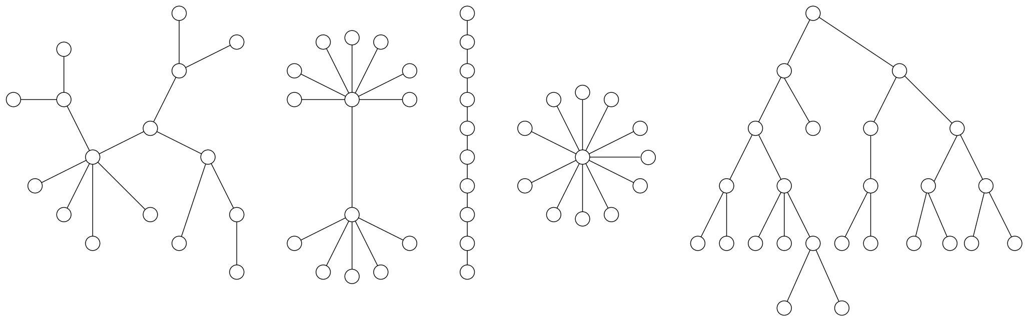

Figure 23: Five trees

10 Trees

We have met trees when studying enumeration problems; now we have a look at them as graphs. A graph $G = (V, E)$ is called a *tree* if it is connected and contains no cycle as a subgraph. The simplest tree has one node and no edges. The second simplest tree consists of two nodes connected by an edge. Figure 23 shows a variety of other trees.

Note that the two properties defining trees work in the opposite direction: connectedness means that the graph cannot have "too few" edges, while the exclusion of cycles means that it cannot have "too many". To be more precise, if a graph is connected, then adding a new edge to it, it remains connected (while deleting an edge, it may or may not stay connected). If a graph contains no cycle, then deleting any edge, the remaining graph will not contain a cycle either (while adding a new edge may or may ot create a cycle). The following theorem shows that trees can be characterized as "minimally connected" graphs as well as "maximally cycle-free" graphs.

Theorem 10.1 *(a) A graph G is a tree if and only if it is connected, but deleting any of its edges results in a disconnected graph.*

(b) A graph G is a tree if and only if it contains no cycles, but adding any new edge creates a cycle.

We prove part (a) of this theorem; the proof of part (b) is left as an exercise.

First, we have to prove that if G is a tree then it satisfies the condition given in the theorem. It is clear that G is connected (by the definition of a tree). We want to prove that deleting any edge, it cannot remain connected. The proof is indirect: assume that deleting the edge uv from a tree G, the remaining graph G' is connected. Then G' contains a path P connecting u and v. But then, if we put the edge uv back, the path P and the edge uv will form a cycle in G, which contradicts the definition of trees.

Second, we have to prove that if G satisfies the condition given in the theorem, then it is a tree. It is clear that G is connected, so we only have to argue that G does not contain a cycle. Again by an indirect argument, assume that G does contain a cycle C. Then deleting any edge of C, we obtain a connected graph (exercise 9.2). But this contradicts the condition in the theorem.

10.1 Prove part (b) of theorem 10.1.

10.2 Prove that connecting two nodes u and v in a graph G by a new edge creates a new cycle if and only if u and v are in the same connected component of G.

10.3 Prove that in a tree, every two nodes can be connected by a *unique* path. Conversely, prove that if a graph G has the property that every two nodes can be connected by a path, and there is only one connecting path for each pair, then the graph is a tree.

10.1 How to grow a tree?

The following is one of the most important properties of trees.

Theorem 10.2 *Every tree with at least two nodes has at least two nodes of degree 1.*

Let G be a tree with at least two nodes. We prove that G has a node of degree 1, and leave it to the reader as an exercise to prove that it has at least one more. (A path has only two such nodes, so this is the best possible we can claim.)

Let us start from any node v_0 of the tree and take a walk (climb?) on the tree. Let's say we never want to turn back from a node on the edge through which we entered it; this is possible unless we get to a node of degree 1, in which case we stop and the proof is finished.

So let's argue that this must happen sooner or later. If not, then eventually we must return to a node we have already visited; but then the nodes and edges we have traversed between the two visits form a cycle. This contradicts our assumption that G is a tree and hence contains no cycle.

10.4 Apply the argument above to find a second node of degree 1.

A real tree grows by developing a new twig again and again. We show that graph-trees can be grown in the same way. To be more precise, consider the following procedure, which we call the *Tree-growing Procedure*:

— *Start with a single node.*

— *Repeat the following any number of times: if you have any graph G, create a new node and connect it by a new edge to any node of G.*

Theorem 10.3 *Every graph obtained by the Tree-growing Procedure is a tree, and every tree can be obtained this way.*

The proof of this is again rather straightforward, but let us go through it, if only to gain practice in arguing about graphs.

First, consider any graph that can be obtained by this procedure. The starting graph is certainly a tree, so it suffices to argue that we never create a non-tree; in other words, if G is a tree, and G' is obtained from G by creating a new node v and connecting it to a node u of G, then G' is a tree. This is straightforward: G' is connected, since any two "old" nodes can be connected by a path in G, while v can be connected to any other node w by first going to u and then connecting u to w. Moreover, G cannot contain a cycle: v has degree 1 and so no cycle can go through v, but a cycle that does not go through v would be a cycle in the old graph, which is supposed to be a tree.

Second, let's argue that every tree can be constructed this way. We prove this by induction on the number of nodes.[4] If the number of nodes is 1, then the tree arises by the construction, since this is the way we start. Assume that G is a tree with at least 2 nodes. Then by theorem

[4]The first part of the proof is also an induction argument, even though it was not phrased as such.

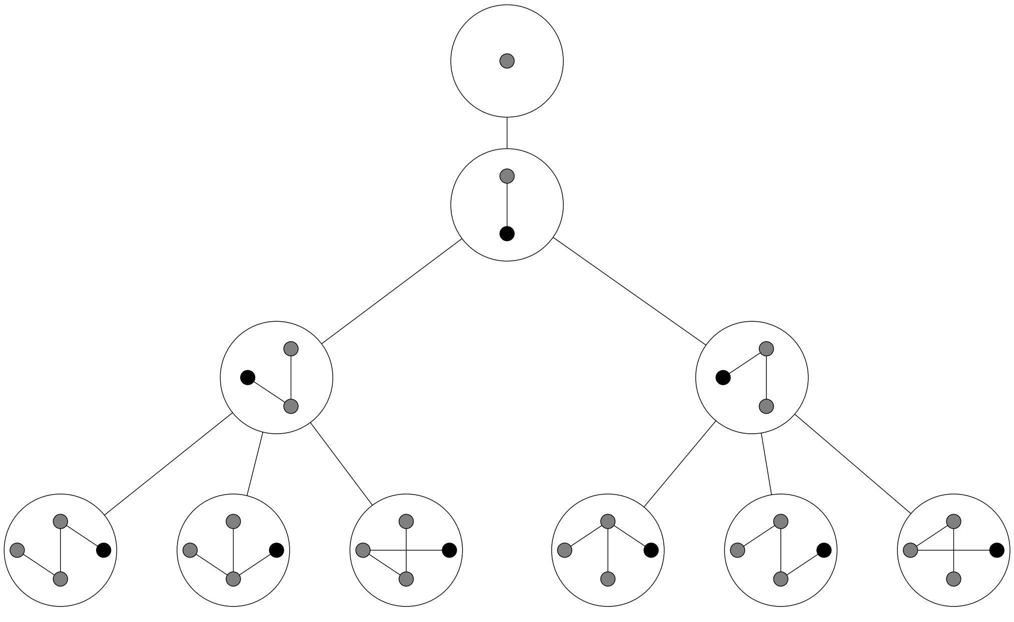

Figure 24: The descendence tree of trees

10.2, G has a node of degree 1 (at least two nodes, in fact). Let v be a node with degree 1. Delete v from G, together with the edge with endpoint v, to get a graph G'.

We claim that G' is a tree. Indeed, G' is connected: any two nodes of G' can be connected by a path in G, and this path cannot go through v as v has degree 1. So this path is also a path in G'. Furthermore, G' does not contain a cycle as G does not.

By the induction hypothesis, every tree with fewer nodes than G arises by the construction; in particular, G' does. But then G arises from G' by one more iteration of the second step. This completes the proof of Theorem 10.3.

Figure 24 shows how trees with up to 4 nodes arise by this construction. Note that there is a "tree of trees" here. The fact that the logical structure of this construction is a tree does not have anything to do with the fact that we are constructing trees: any iterative construction with free choices at each step results in a similar "descendence tree".

The Tree-growing Procedure can be used to establish a number of properties of trees. Perhaps most important of these concerns the number of edges. How many edges does a tree have? Of course, this depends on the number of nodes; but surprisingly, it depends *only* on the number of nodes:

Theorem 10.4 *Every tree on n nodes has $n-1$ edges.*

Indeed, we start with one more node (1) than edge (0), and at each step, one new node and one new edge is added, so this difference of 1 is maintained.

> **10.5** Let G be a tree, which we consider as the network of roads in a medieval country, with castles as nodes. The King lives at node r. On a certain day, the lord of each castle sets out to visit the King. Argue carefully that soon after they leave their castles, there will be exactly one lord on each edge. Give a proof of Theorem 10.4 based on this.

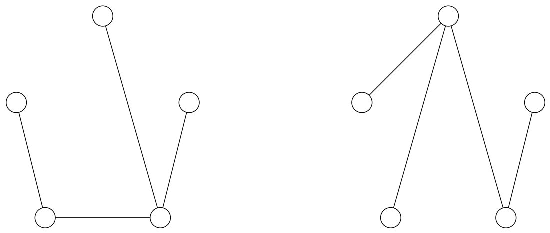

Figure 25: Are these trees the same?

10.6 If we delete a node v from a tree (together with all edges that end there), we get a graph whose connected components are trees. We call these connected components the *branches* at node v. Prove that every tree has a node such that every branch at this node contains at most half the nodes of the tree.

10.2 Rooted trees

Often we use trees that have a special node, which we call the *root.* For example, trees that occured in counting subsets or permutations were built starting with a given node.

We can take any tree, select any of its nodes, and call it a *root.* A tree with a specified root is called a *rooted tree.*

Let G be a rooted tree with root r. Given any node v different from r, we know that the tree contains a unique path connecting v to r. The node on this path next to v is called the *father* of v. The other neighbors of v are called the *sons* of v. The root r does not have a father, but all its neighbors are called its sons.

Now a basic geneological assertion: *every node is the father of its sons.* Indeed, let v be any node and let u be one of its sons. Consider the unique path P connecting v to r. The node cannot lie on P: it cannot be the first node after v, since then it wold be the father of v, and not its son; and it cannot be a later node, since then going from v to u on the path P and then back to v on the edge uv we would traverse a cycle. But this implies that adding the node u and the edge uv to P we get a path connecting u to r. Since v is the forst node on this path after u, it follows that v is the father of u. (Is this argument valid when $v = r$? Check!)

We have seen that every node different from the root has exactly one father. A node can have any number of sons, including zero. A node with no sons is called a *leaf.* In other words, a leaf is a node with degree 1, different from r. (To be precise, if the tree consists of a single node r, then this is a leaf.)

10.3 How many trees are there?

We have counted all sorts of things in the first part of this book; now that we are familiar with trees, it is natural to ask: *how many trees are there on n nodes?*

Before attempting to answer this question, we have to clarify an important issue: when do we consider two trees different? There are more than one reasonable answers to this question. Consider the trees in Figure 25. Are they the same? One could say that they are; but then, if the nodes are, say, towns, and the edges represent roads to be built between them, then clearly the inhabitants of the towns will consider the two plans very different.

So we have to define carefully when we consider two trees the same. The following are two possibilities:

— We fix the set of nodes, and consider two trees the same if the same pairs of nodes are connected in each. (This is the position the twon people would take when they consider road construction plans.) In this case, it is advisable to give names to the nodes, so that we can distinguish them. It is convenient to use the numbers $0, 1, 2, \ldots, n-1$ as names (if the tree has n nodes). We express this by saying that the vertices of the tree are labeled by $0, 1, 2, \ldots n-1$. Figure 26 shows a labelled tree. Interchanging the labels 2 and 4 (say) would yield a different labelled tree.

— We don't give names to the nodes, and consider two trees the same if we can rearrange the nodes of one so that we get the other tree. More exactly, we consider two trees the same (the mathematical term for this is *isomorphic*) if there exists a one-to-one correpondence between the nodes of the first tree and the nodes of the second tree so that two nodes in the first tree that are connected by an edge correspond to nodes in the second tree that are connected by an edge, and vice versa. If we speak about *unlabelled trees*, we mean that we don't distinguish isomorphic trees from each other. For example, all paths on n nodes are the same as unlabelled trees.

So we can ask two questions: how many labelled trees are there on n nodes? and how many unlabelled trees are there on n nodes? These are really two different questions, and we have to consider them separately.

10.7 Find all unlabelled trees on $2, 3, 4$ and 5 nodes. How many labelled trees do you get from each? Use this to find the number of labelled trees on $2, 3, 4$ and 5 nodes.

10.8 How many labelled trees on n nodes are stars? How many are paths?

The number of labelled trees. For the case of labelled trees, there is a very nice solution.

Theorem 10.5 (Cayley's Theorem) *The number of labeled trees on n nodes is n^{n-2}.*

The formula is elegant, but the surprising fact about it is that it is quite difficult to prove! It is substantially deeper than any of the previous formulas for the number of this and that. There are various ways to prove it, but each uses some deeper tool from mathematics or a deeper idea. We'll give a proof that is perhaps best understood by first discussing a quite different question in computer science: how to store a tree?

10.4 How to store a tree?

Suppose that you want to store a labelled tree, say the tree in Figure 26, in a computer. How would you do this? Of course, the answer depends on what you need to store the tree for, what information about it you want to retrieve and how often, etc. Right now, we are only concerned with the amount of memory we need. We want to store the tree so that it occupies the least amount of memory.

Let's try some simple solutions.

(a) Suppose that we have a tree G with n nodes. One thing that comes to mind is to make a big table, with n rows and n columns, and put (say) the number 1 in the j-th position of

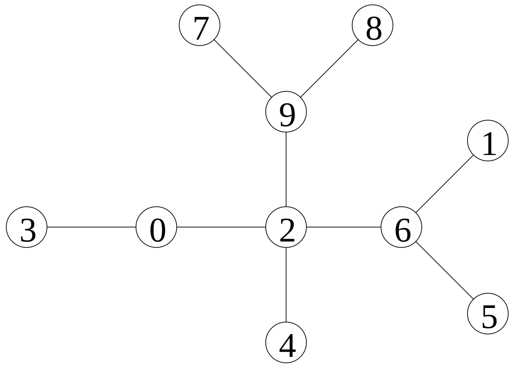

Figure 26: A labelled tree

the i-th row if nodes i and j are connected by an edge, and the number 0, if they are not:

$$\begin{array}{cccccccccc}
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array} \tag{20}$$

We need one bit to store each entry of this table, so this takes n^2 bits. We can save a little by noticing that it is enough to store the part below the diagonal, since the diagonal is always 0 and the other half of the table is just the reflection of the half below the diagonal. But this is still $(n^2 - n)/2$ bits.

This method of storing the tree can, of course, be used for any graph. It is often very useful, but, at least for trees, it is very wasteful.

(b) We fare better if we specify each tree by listing all its edges. We can specify each edge by its two endpoints. It will be convenient to arrange this list in an array whose columns correspond to the edges. For example, the tree in Figure 26 can be encoded by

$$\begin{array}{ccccccccc}
7 & 8 & 9 & 6 & 3 & 0 & 2 & 6 & 6 \\
9 & 9 & 2 & 2 & 0 & 2 & 4 & 1 & 5
\end{array}$$

Instead of a table with n rows, we get a table just with two rows. We pay a little for this: instead of just 0 and 1, the table wil contain integers between 0 and $n-1$. But this is certainly worth it: even if we count bits, to write down the label of a node takes $\log_2 n$ bits, so the whole table occupies only $2n \log_2 n$ bits, which is much less than $(n^2 - n)/2$ if n is large.

There is still a lot of free choice here, which means that the same tree may be encoded in different ways: we have freedom in choosing the order of the edges, and also in choosing the order in which the two endpoints of an edge are listed. We could agree on some arbitrary conventions to make the code well defined (say, listing the two endnodes of an edge in increasing order, and then the edges in increasing order of their first endpoints, breaking ties according to the second endpoints); but it will be more useful to do this in a way that also allows us to save more memory.

(c) **The father code.** From now on, the node with label 0 will play a special role; we'll consider it the "root" of the tree. Then we can list the two endnodes of an edge by listing the endpoint further from the root first, and then the endpoint nearer to the root second. So for every edge, the node written below is the father of the node written above. For the order in which we list the edges, let us take the order of their first nodes. For the tree in Figure 26, we get the table

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

Do you notice anything special about this table? The first row consists of the numbers $1, 2, 3, 4, 5, 6, 7, 8, 9$, in this order. Is this a coincidence? Well, the order is certainly not (we ordered the edges by the increasing order of their first endpoints), but why do we get every number exactly once? After a little reflection, this should be also clear: if a node occurs in the first row, then its father occurs below it. Since a node has only one father, it can occur only once. Since every node other than the root has a father, every node other than the root occurs in the first row.

Thus we know in advance that if we have a tree on n nodes, and write up the array using this method, then the first row will consist of $1, 2, 3, \dots, n-1$. So we may as well suppress the first row without loosing any information; it suffices to store the second. Thus we can specify the tree by a sequence of $n-1$ numbers, each between 0 and $n-1$. This takes $(n-1)\lceil \log_2 n \rceil$ bits.

This coding is not optimal, in the sense that not every "code" gives a tree (see exercise 10.4). But we'll see that this method is already nearly optimal.

10.9 Consider the following "codes": $(0, 1, 2, 3, 4, 5, 6, 7)$; $(7, 6, 5, 4, 3, 2, 1, 0)$; $(0, 0, 0, 0, 0, 0, 0, 0)$; $(2, 3, 1, 2, 3, 1, 2, 3)$. Which of these are "father codes" of trees?

10.10 Prove, based on the "father code" method of storing trees, that the number of labelled trees on n nodes is at most n^{n-1}.

(d) Now we describe a procedure, the so-called *Prüfer code*, that will assign to any n-point labeled tree a sequence of length $n-2$, not $n-1$, consisting of the numbers $0, \dots, n-1$. The gain is little, but important: we'll show that every such sequence corresponds to a tree. Thus we will establish a *bijection*, a one-to-one correspondence between labelled trees on n nodes and sequences of length $n-2$, consisting of numbers $0, 1, \dots, n-1$. Since the number of such sequences is n^{n-2}, this will also prove Cayley's Theorem.

The Prüfer code can be considered as a refinement of method (c). We still consider 0 as the root, we still order the two endpoints of an edge so that the father comes first, but we order the edges (the columns of the array) not by the magnitude of their first endpoint but a little differently, more closely related to the tree itself.

So again, we construct a table with two rows, whose columns correspond to the edges, and each edge is listed so that the node farther from 0 is on the top, its father on the bottom. The issue is the order in which we list the edges.

Here is the rule for this order: we look for a node of degree 1, different from 0, with the smallest label, and write down the edge with this endnode. In our example, this means that we write down $\frac{1}{6}$. Then we delete this node and edge from the tree, and repeat: we look for the endnode with smallest label, different from 0, and write down the edge incident with it. In our case, this means adding a column $\frac{3}{0}$ to the table. Then we delete this node and edge etc. We go until all edges are listed. The array we get is called the *extended Prüfer code* of the tree (we call it extended because, as we'll see, we only need a part of it as the "real" Prüfer code). The extended Prüfer code of the tree in Figure 26 is:

$$\begin{array}{ccccccccc} 1 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 2 \\ 6 & 0 & 2 & 6 & 2 & 9 & 9 & 2 & 0 \end{array}$$

Why is this any better than the "father code"? One little observation is that the last entry in the second row is now always 0, since it comes from the last edge and since we never touched the node 0, this last edge must be incident with it. But we have payed a lot for this, it seems: it is not clear any more that the first row is superfluous; it still consists of the numbers $1, 2, \ldots, n-1$, but now they are not in increasing order any more.

The key lemma is that the first row is determined by the second:

Lemma 10.1 *The second row of an extended Prüfer code determines the first.*

Let us illustrate the proof of the lemma by an example. Suppose that somebody gives us the second row of an extended Prüfer code of a labelled tree on 8 nodes; say, 2403310 (we have one fewer edges than nodes, so the second row consists of 7 numbers, and as we have seen, it must end with a 0). Let us figure out what the first row must have been.

How does the first row start? Remember that this is the node that we delete in the first step; by the rule of cinstructing the Prüfer code, this is the node with degree 1 with smallest label. Could this node be the node 1? tree? No, because then we would have to delete it in the first step, and it could not occur any more, but it does. By the same token, no number occuring in the second row could be a leaf of the tree at the beginning. This rules out 2, 3 and 4.

What about 5? It does not occur in the second row; does this mean that it is a leaf of the original tree? The answer is yes; else, 5 would have been the father of some other node, and it would have been written in the second row when this other node was deleted. Thus 5 was its leaf with smallest label, and the first row of the extended Prüfer code must start with 5.

Let's try to figure out the next entry in the first row, which is, as we know, the leaf with smallest label of the tree after 5 was deleted. The node 1 is still ruled out, since it occurs later in the second row later; but 2 does not occur any more, which means (by the same argument as before), that 2 was the leaf with smallest label after deleting 5. Thus the second entry in the first row is 2.

Similarly, the third entry must be 4, since all the smaller numbers either occur later or have been used already. Going on in a similar fashion, we get that the full array must have been:

$$\begin{array}{ccccccc} 5 & 2 & 4 & 6 & 7 & 3 & 1 \\ 2 & 4 & 0 & 3 & 3 & 1 & 0 \end{array}$$

This corresponds to the tree in Figure 27

The considerations above are completely general, and can be summed up as follows:

each entry in the first row of the extended Prüfer code is the smallest integer that does not occur in the first row before it, nor in the second row below or after it.

Indeed, when this entry (say, the k-th entry in the first row) was recorded, then the nodes before it in the first row were deleted (together with the edges corresponding to the first $k-1$

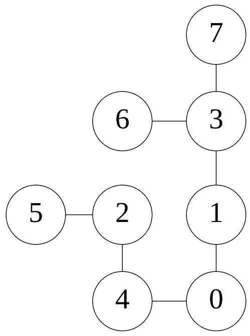

Figure 27: A tree reconstructed from its Prüfer code

columns). The remaining entries in the second row are exactly those nodes that are fathers at this time, which means that they are not leaves.

This describes how the first row can be reconstructed from the second. So we don't need the full extended Prüfer code to store the tree; it suffices to store the second row. In fact, we know that the last entry in the second row is n, so we don't have to store this either. The sequence consisting of the first $n-2$ entries of the second row is called the *Prüfer code* of the tree. Thus the Prüfer code is a sequence of length $n-2$, each entry of which is a number between 0 and $n-1$.

This is similar to the father code, just one shorter; not much gain here for all the work. But the beauty of the Prüfer code is that it is optimal, in the sense that

every sequence of numbers between 0 *and* $n-1$*, of length* $n-2$*, is a Prüfer code of some tree on* n *nodes.*

This can be proved in two steps. First, we extend this sequence to a table with two rows: we add an n at the end, and then write above each entry in the first row the smallest integer that does not occur in the first row before it, nor in the second row below or after it (note that it is always possible to find such an integer: the condition excludes at most $n-1$ values out of n).

Now this table with two rows is the Prüfer code of a tree. The proof of this fact, which is not difficult any more, is left to the reader as an exercise.

10.11 Complete the proof.

Let us sum up what the Prüfer code gives. First, it proves Cayley's Theorem. Second, it provides a theoretically most efficient way of encoding trees. Each Prüfer code can be considered as a natural number written in the base n number system; in this way, we order a "serial number" between 0 and n^{n-2} to the n-point labeled trees. Expressing these serial numbers in base two, we get a code by $0-1$ sequences in of length at most $\lceil (n-2)\log n \rceil$.

As a third use of the Prüfer code, let's suppose that we want to write a program that generates a random labeled tree on n nodes in such a way that all trees occur with the same probability. This is not easy from scratch; but the Prüfer code gives an efficient solution. We just have to generate $n-2$ independent random integers between 0 and $n-1$ (most programming languages have a statement for this) and then "decode" this sequence as a tree, as described.

The number of unlabelled trees. The number of unlabelled trees on n nodes, usually

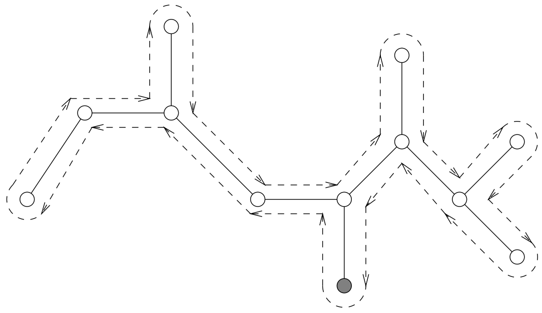

Figure 28: Walking around a tree

denoted by T_n, is even more difficult to handle. No simple formula like Cayley's theorem is known for this number. Our goal is to get a rough idea about how large this number is.

There is only one unlabelled tree on $1, 2$ or 3 nodes; there are two on 4 nodes (the path and the star). There are 3 on 5 nodes (the star, the path, and the tree in Figure 25. These numbers are much smaller than the number of labelled trees with these numbers of nodes, which are $1, 1, 3, 16$ and 125 by Cayley's Theorem.

It is of course clear that the number of unlabelled trees is less than the number of labelled trees; every unlabelled tree can be labelled in many ways. How many ways? If we draw an unlabelled tree, we can label its nodes in $n!$ ways. The labelled trees we get this way are not necessarily all different: for example, if the tree is a "star", i.e., it $n-2$ leaves and one more node connected to all leaves, then no matter how we permute the labels of the leaves, we get the same labelled tree. So an unlabelled star yields n labelled stars.

But at least we know that each labeled tree can be labeled in at most $n!$ ways. Since the number of labelled trees is n^{n-2}, it follows that the number of unlabeled trees is at least $n^{n-2}/n!$. Using Stirling's formula (Theorem 2.5), we see that this number is about $e^n/n^{5/2}\sqrt{2\pi}$.

This number is musch smaller than the number of labelled trees, n^{n-2}, but of course it is only a *lower bound* on the number of unlabelled trees. How can we obtain an *upper bound* on this number? If we think in terms of storage, the issue is: can we store an unlabelled tree more economically than labelling its nodes and then storing it as a labelled tree? Very informally, how should we describe a tree, if we only want the "shape" of it, and don't care which node gets which label?

Take an n-point tree G, and specify one of its leaves as its "root". Next, draw G in the plane without crossing edges; this can always be done, and we almost always draw trees this way.

Now we imagine that the edges of the tree are walls, perpendicular to the plane. Starting at the root, walk around this system of walls, keeping the wall always to your right. We'll call walking along an edge a "step". Since there are $n-1$ edges, and each edge has two sides, we'll make $2(n-1)$ steps before returning to the root (Figure 28).

Each time we make a step *away* from the root (i.e., a step from a father to one of its sons), we write down a 1; each time we make a step *toward* the root, we write down a 0. This way we end up with a sequence of length $2(n-1)$, consisting of 0's and 1's. We call this sequence the *planar code* of the (unlabelled) tree. The planar code of the tree in figure 28 is 1110011010100100.

Now this name already indicates that the planar code has the following important property:

Every unlabelled tree is uniquely determined by its planar code.

Let us illuminate the proof of this by assuming that the tree is covered by snow, and we only have its code. We ask a friend of ours to walk around the tree just like above, and uncover the walls, and we look at the code in the meanwhile. What do we see? Any time we see a 1, he walks along a wall away from the root, and he cleans it from the snow. We see this as growing a new twig. Any time we see a 0, he walks back, along an edge already uncovered, toward to root.

Now this describes a perfectly good way to draw the tree: we look at the bits of the code one by one, while keeping the pen on the paper. Any time we see a 1, we draw a new edge to a new node (and move the pen to the new node). Any time we see a 0, we move the pen back by one edge toward the root. Thus the tree is indeed determined by its planar code.

Since the number of possible planar codes is at most $2^{2(n-1)} = 4^{n-1}$, we get that the number of unlabelled trees is at most this large. Summing up:

Theorem 10.6 *The number T_n of unlabelled trees with n nodes satisfies*

$$\frac{n^{n-2}}{n!} < T_n < 4^{n-1}.$$

The exact form of this lower bound does not matter much; we can conclude, just to have a statement simpler to remember, that the number of unlabelled trees on n nodes is larger than 2^n if n is large enough ($n > 30$ if you work it out). So we get, at least for $n \geq 30$, the following bounds that are easier to remember:

$$2^n \leq T_n \leq 4^n.$$

The planar code is far from optimal; every unlabelled tree has many different codes (depending on how we draw it in the plane and how we choose the root), and not every 0-1 sequence of length $2(n-1)$ is a code of a tree (for example, it must start with a 1 and have the same number of 0's as 1's). Still, the planar code is quite an efficient way of encoding unlabelled trees: it uses less than $2n$ bits for trees with n nodes. Since there are more than 2^n unlabelled trees (at least for $n > 30$), we could not possibly get by with codes of length n: there are just not enough of them.

Unlike for labelled trees, we don't know a simple formula for the number of unlabelled trees on n nodes, and probably none exists. According to a difficult result of George Pólya, the number of unlabelled trees on n nodes is asymptotically $an^{3/2}b^n$, where a and b are real numbers defined in a complicated way.

10.12 Does there exist an unlabelled tree with planar code (a) 1111111100000000; (b) 1010101010101010; (c) 1100011100?

11 Finding the optimum

11.1 Finding the best tree

A country with a n towns wants to construct a new telephone network to connect all towns. Of course, they don't have to build a separate line between every pair of towns; but they do need to build a connected network; in our terms, this means that the graph of direct connections must form a connected graph. Let's assume that they don't want to build a direct line between town that can be reached otherwise (there may be good reasons for doing so, as we shall see later, but at the moment let's assume their only goal is to get a connected network). Thus they want to build a minimal connected graph with these nodes, i.e., a tree.

We know that no matter which tree they choose to build, they have to construct $n-1$ lines. Does this mean that it does not matter which tree they build? No, because lines are not equally easy to build. Some lines between towns may cost much more than some other lines, depending on how far the towns are, whether there are mountains or lakes between them etc. So the task is to find a spanning tree whose total cost (the sum of costs of its edges) is minimal.

How do we know what these costs are? Well, this is not something mathematics can tell you; it is the job of the engineers and economisits to estimate the cost of each possible line in advance. So we just assume that these costs are given.

At this point, the tasks seems trivial (very easy) again: just compute the cost of each tree on these nodes, and select the tree with smallest cost.

We dispute the claim that this is easy. The number of trees to consider is enormous: We know (by Cayley's Theorem 10.5) that the number of labelled trees on n nodes is n^{n-2}. So for 10 cities, we'd have to look at 10^8 (hundred million) possible trees; for 20 cities, the number is astronomical (more than 10^{20}). We have to find a better way to select an optimal tree; and that's the point where mathematics comes to the rescue.

There is this story about the pessimist and the optimist: they both get a box of assorted candies. The optimist always picks the best; the pessimist always eats the worst (to save the better candies for later). So the optimist always eats the best available candy, the pessimist always eats the worst available candy — and yet, they end up with eating exactly the same.

So let's see how the optimistic government would build the telephone network. They start with raising money; as soon as they have enough money to build a line (the cheapest line), they build it. Then they wait until they have enough money to build a second connection. Then they wait until they have enough money to build a third connection... It may happen that the third cheapest connection forms a triangle with the first two (say, three towns are close to each other). Then, of course, they skip this and raise enough money to build the fourth cheapest connection.

At any time, the optimistic government will wait until they have enough money to build a connection between two towns that are not yet connected by any path, and build this connection.

Finally, they will get a connected graph on the n nodes representing the towns. The graph does not contain a cycle, since the edge of the cycle constructed last would connect two towns that are already accessible from each other through the other edges of the cycle. So, the graph they get is indeed a tree.

But is this network the cheapest possible? Could stinginess at the beginning backfire and force them to spend much more at the end? We'll prove below that our optimistic government

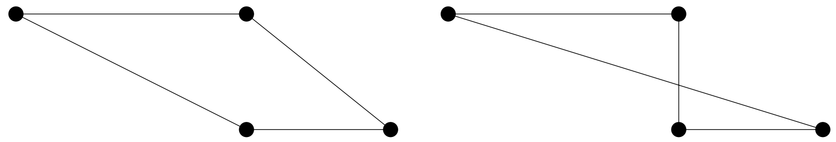

Figure 29: Failure of the greedy method. Construction costs are proportional to the distance. The first figure shows a cheapest (shortest) cycle through all 4 towns; the second shows the cycle obtained by the optimistic (greedy) method.

has undeserved success: the tree they build is as cheap as possible.

Before we jump into the proof, we should discuss why we said that the government's success was "undeserved". We show that if we modify the task a little, the same optimistic approach might lead to very bad results.

Let us assume that, for reasons of reliability, they require that between any two towns, there should be at least two paths with no edge in common (this guarantees that when a line if inoperational because of failure or maintanance, any two towns can still be connected). For this, $n-1$ lines are not enough ($n-1$ edges forming a connected graph must form a tree, but then deleting any edge, the rest will not be connected any more). But n lines suffice: all we have to do is to draw a single cycle through all the towns. This leads to the following task:

Find a cycle with given n towns as nodes, so that the total cost of constructing its edges is minimum.

(This problem is one of the most famous tasks in mathematical optimization: it is called the Travelling Salesman Problem. We'll say more about it later.)

Our optimistic government would do the following: build the cheapest line, then the second cheapest, then the third cheapest etc, skipping the construction of lines that are superfluous: it will not build a third edge out of a town that already has two, and will not build an edge completing a cycle unless this cycle connects all nodes. Eventually they get a cycle through all towns, but this is not necessarily the best! Figure 29 shows an example where the optimistic method (called "greedy" in this area of applied mathematics) gives a cycle that is quite a bit worse than optimal.

So the greedy method can be bad for the solution of a problem that is only slightly different from the problem of finding the cheapest tree. Thus the fact (to be proved below) that the optimistic governments builds the best tree is indeed undeserved luck.

So let us return to the solution of the problem of finding a tree with minimum cost, and prove that the optimistic method yields a cheapest tree. Let us call the tree obtained by the greedy method the *greedy tree*, and denote it by F. In other words, we want to prove that any other tree would cost at least as much as the greedy tree (and so no one could accuse the government of wasting money, and justify the accusation by exhibiting another tree that would have been cheaper).

So let G be any tree different from the greedy tree F. Let us imagine the process of contructing F, and the step when we first pick an edge that is not an edge of G. Let e be this edge. If we add e to G, we get a cycle C. This cycle is not fully contained in F, so it has an edge f that is not an edge of F (Figure 30). If we add the edge e to G and then delete f, we get a (third) tree H. (Why is H a tree? Give an argument!)

We want to show that H *is at most as expensive as* G. This clearly means that e is at most as expensive as f. Suppose (by indirect argument) that f is cheaper than e.

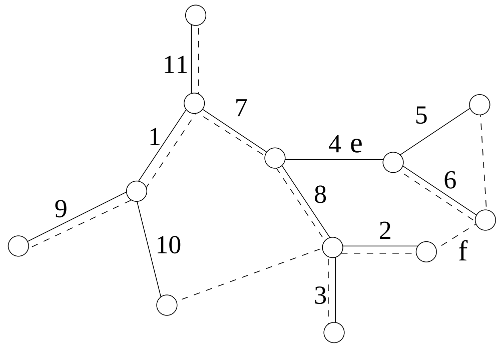

Figure 30: The greedy tree is optimal

Now comes a crucial question: why didn't the optimistic government select f instead of e at this point in time? The only reason could be that f was ruled out because it would have formed a cycle C' with the edges of F already selected. But all these previously selected edges are edges of G, since we are inspecting the step when the first edge not in G was added to F. Since f itself is an edge of G, it follows that all edges of C' are edges of G, which is impossible since G is a tree. This contradiction proves that f cannot be cheaper than e and hence G cannot be cheaper than H.

So we replace G by this tree H that is not more expensive. In addition, the new tree H has the adventage that coincides with F in more edges, since we deleted from G an edge not in F and added an edge in F. This implies that if H is different from F and we repeat the same argument again and again, we get trees that are not more expensive than G, and coincide with F in more and more edges. Sooner of later we must end up with F itself, proving that F was no more expensive than G.

11.1 A pessimistic government could follow the following logic: if we are not careful, we may end up with having to build that extremely expensive connection through the mountain; so let us decide right away that building this connection is not an option, and mark it as "impossible". Similarly, let us find the second most expensive line and mark it "impossible", etc. Well, we cannot go on like this forever: we have to look at the graph formed by those edges that are still possible, and this "possiblity graph" must stay connected. In other words, if deleting the most expensive edge that is still possible from the possibility graph would destroy the connectivity of this graph, then like it or not, we have to build this line. So we build this line (the pessimistic government ends up building the most expensive line among those that are still possible). Then they go on to find the most expensive line among those that are still possible and not yet built, mark it impossible if this does not disconnect the possibility graph etc.

Prove that the pessimistic government will have the same total cost as the optimistic.

11.2 In a more real-life government, optimists and pessimists win in unpredictable order. This means that sometimes they build the cheapest line that does not create a cycle with those lines already constructed; sometimes they mark the most expensive lines "impossible" until they get to a line that cannot be marked impossible without disconnecting the network, and then they build it. Prove that they still end up with the same cost.

11.3 If the seat of the government is town r, then quite likely the first line constructed will be the cheapest line out of r (to some town s, say), then the cheapest line that connects either r or s to a new town etc. In general, there will be a connected graph of telephone lines constructed on a subset S of the towns including the capital, and the next line will be the cheapest among all lines that connect S to a node outside S. Prove that the lucky government still obtains a cheapest possible tree.

11.4 Formulate how the pessimistic government will construct a cycle through all towns. Show by an example that they don't always get the cheapest solution.

11.2 Traveling Salesman

Let us return to the question of finding a cheapest possible cycle through all the given towns: we have n towns (points) in the plane, and for any two of them we are given the "cost" of connecting them directly. We have to find a cycle with these nodes so that the cost of the cycle (the sum of the costs of its edges) is as small as possible.

This problem is one of the most important in the area of *combinatorial optimization*, the field dealing with finding the best possible design in various combinatorial situations, like finding the optimal tree discussed in the previous section. It is called the TRAVELLING SALESMAN PROBLEM, and it appears in many disguises. Its name comes from the version of the problem where a travelling salesman has to visit all towns in the region and then return to his home, and of course he wants to minimize his travel costs. It is clear that mathematically, this is the same problem. It is easy to imagine that one and the same mathematical problem appears in connection with designing optimal delivery routes for mail, optimal routes for garbage collection etc.

The following important question leads to the same mathematical problem, except on an entirely different scale. A machine has to drill a number of holes in a printed circuit board (this number could be in the thousands), and then return to the starting point. In this case, the important quantity is the *time* it takes to move the drilling head from one hole to the next, since the total time a given board has to spend on the machine determines the number of boards that can be processed in a day. So if we take the time needed to move the head from one hole to another as the "cost" of this edge, we need to find a cycle with the holes as nodes, and with minimum cost.

The Travelling Salesman Problem is much more difficult than the problem of finding the cheapest tree, and there is no algorithm to solve it that would be anywhere nearly as simple, elegant and efficient as the "optimistic" algorithm discussed in the previous section. There are methods that work quite well most of the time, but they are beyond the scope of this book.

But we want to show at least one simple algorithm that, even though it does not give the best solution, never looses more than a factor of 2. We describe this algorithm in the case when the cost of an edge is just its length, but it would not make any difference to consider any other measure (like time, or the price of a ticket), at least as long as the costs $c(ij)$ satisfy the *triangle inequality*:

$$c(ij) + c(jk) \geq c(ik)$$

(Air fares sometimes don't satisfy this inequality: it may be cheaper to fly from New York to Chicago to Philadelphia then to fly from New York to Philadelphia. But in this case, of course, we might consider the flight New York–Chicago–Philadelphia as one "edge", which does not count as a visit in Chicago.)

We begin by solving a problem we know how to solve: find a cheapest tree connecting up the given nodes. We can use any of the algorithms discussed in the previous section for this. So we find the cheapest tree T, with total cost c.

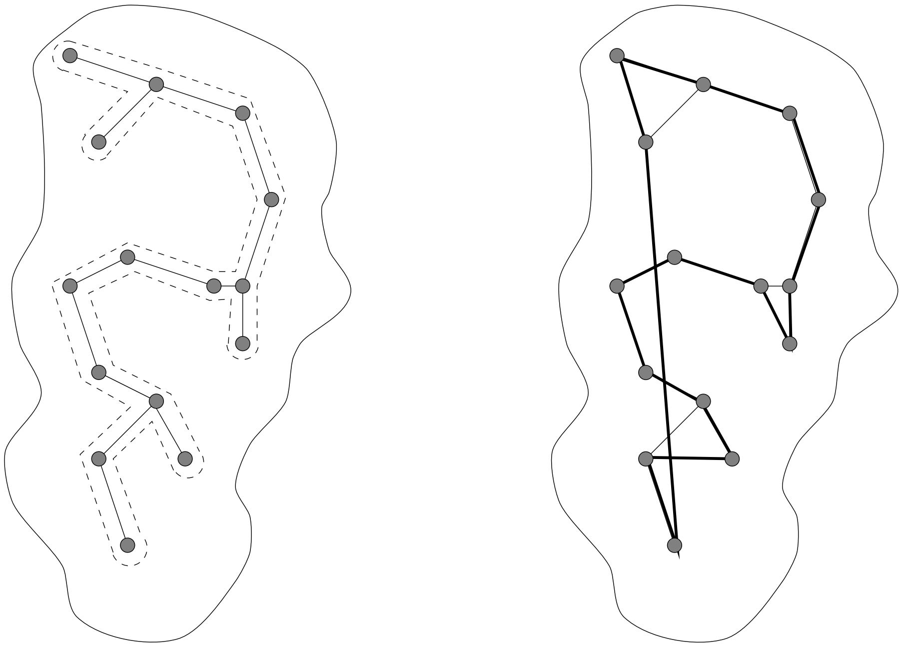

Figure 31: The cheapest tree connecting 15 given towns, the walk around it, and the tour produced by shortcuts. Costs are proportional to distances.

Now how does this tree help in finding a tour? One thing we can do is to walk around the tree just like we did when constructing the "planar code" of a tree in the proof of theorem 10.6 (see figure 28). This certainly gives a walk through each town, and returns to the starting point. The total cost of this walk is exactly twice the cost c of T, since we used every edge twice.

Of course this walk may pass through some of the towns more than once. But this is just good for us: we can make shortcuts. If the walk takes us from i to j to k, and we have seen j alread, we can proceed directly from i to k. The triangle inequality guarantees that we have only shortened our walk by doing so. Doing such shortcuts as long as we can, we end up with a tour through all the towns whose cost is not more than twice the cost of the cheapest spanning tree (Figure 31).

But we want to relate the cost of the tour we obtained to the cost of the optimum tour, not to the cost of the optimum spanning tree! Well, this is easy now: *the cost of a cheapest spanning tree is always less than the cost of the cheapest tour.* Why? Because we can omit any edge of the cheapest tour, to get a spanning tree. This is a very special kind of tree (a path), and as a spanning tree it may or may not be optimal. However, its cost is certainly not smaller than the cost of the cheapest tree, but smaller than the cost of the optimal tour, which proves the assertion above.

To sum up, the cost of the tour we constructed is at most twice that of the cheapest spanning tree, which in turn is less than twice the cost of a cheapest tour.

11.5 Is the tour in figure 31 shortest possible?

11.6 Prove that if all costs are proportional to distances, then a shortest tour cannot intersect itself.

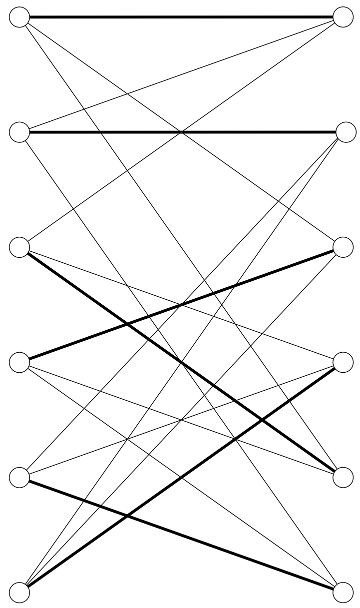

Figure 32: A bipartite graph and a perfect matching in it

12 Matchings in graphs

12.1 A dancing problem

At the prom, 300 students took part. They did not all know each other; in fact, every girl new exactly 50 boys and every boy new exactly 50 girls (we assume, as before, that acquaintance is mutual).

We claim that they can all dance simultaneously (so that only pairs who know each other dance with each other).

Since we are talking about acquaintances, it is natural to describe the situation by a graph (or at least, imagine the graph that describes it). So we draw 300 nodes, each representing a student, and connect two of them if they know each other. Actually, we can make the graph a little simpler: the fact that two boys, or two girls, know each other plays no role whatsoever in this problem: so we don't have to draw those edges that correspond to such acquaintances. We can then arrange the nodes, conveniently, so that the nodes representing boys are on the left, nodes representing girls are on the right; then every edge will connect a node from the left to a node from the right. We shall denote the set of nodes on the left by A, the set of nodes on the right by B.

This way we got a special kind of graph, called a *bipartite* graph. Figure 32 shows such a graph (of course, depicting a smaller party). The thick edges show one way to pair up people for dancing. Such a set of edges is called a *perfect matching.*

To be precise, let's give the definitions of these terms: a graph is *bipartite* if its nodes can be partitioned into two classes, say A and B so that every edge connects a node in A to a node in B. A *perfect matching* is a set of edges such that every node is incident with exactly one of them.

After this, we can formulate our problem in the language of graph theory as follows: we have a bipartite graph with 300 nodes, in which every node has degree 50. We want to prove that it contains a perfect matching.

As before, it is good idea to generalize the assertion to any number of nodes. Let's be daring and guess that the numbers 300 and 50 play no role whatsoever. The only condition that matters is that all nodes have the same degree (and this is not 0). Thus we set out to prove the following theorem:

Theorem 12.1 *If every node of a bipartite graph has the same degree $d \geq 1$, then it contains a perfect matching.*

Before proving the theorem, it will be useful to solve some exercises, and discuss another problem.

12.1 It is obvious that for a bipartite graph to contain a perfect matching, it is necessary that $|A| = |B|$. Show that if every node has the same degree, then this is indeed so.

12.2 Show by examples that the conditions formulated in the theorem cannot be dropped:

(a) A non-bipartite graph in which every node has the same degree need not contain a perfect matching.

(b) A bipartite graph in which every node has positive degree (but not all the same) need not contain a perfect matching.

12.3 Prove Theorem 12.1 for $d = 1$ and $d = 2$.

12.2 Another matching problem

An island is inhabited by six tribes. They are on good terms and split up the island between them, so that each tribe has a hunting territory of 100 square miles. The whole island has an area of 600 miles.

The tribes decide that they all should choose new totems. They decide that each tribe should pick one of the six species of tortoise that live on the island. Of course, they want to pick different totems, and so that the totem fo each tribe should occur somewhere on their territory.

It is given that the areas where the different species of tortoises live don't overlap, and they have they same area - 100 square miles (so it aso follows that some tortoise lives everywhere on the island). Of course, the way the tortoises divide up the islands may be entirely different from the way way the tribes do (Figure 33)

We want to prove that such a selection of totems is always possible.

To see the significance of the conditions, let's assume that we did not stipulate that the area of each tortoise species is the same. Then some species could occupy more - say, 200 square miles. But then it could happen that two of tribes are living on exactly these 200 square miles, and so their only possible choice for a totem would be one and same species.

Let's try to illustrate our problem by a graph. We can represent each tribe by a node, and also each species of tortoise by a node. Let us connect a tribe-node to a species-node is the species occurs somewhere on the territory of the tribe (we could also say that he tribe occurs on the territory of the species — just in case the tortoises want to pick totems too). Drawing the tribe-nodes on the right, and the species-nodes on the left, makes it clear that we get a bipartite graph (Figure 34. And what is it that we want to prove? It is that this graph has a perfect matching!

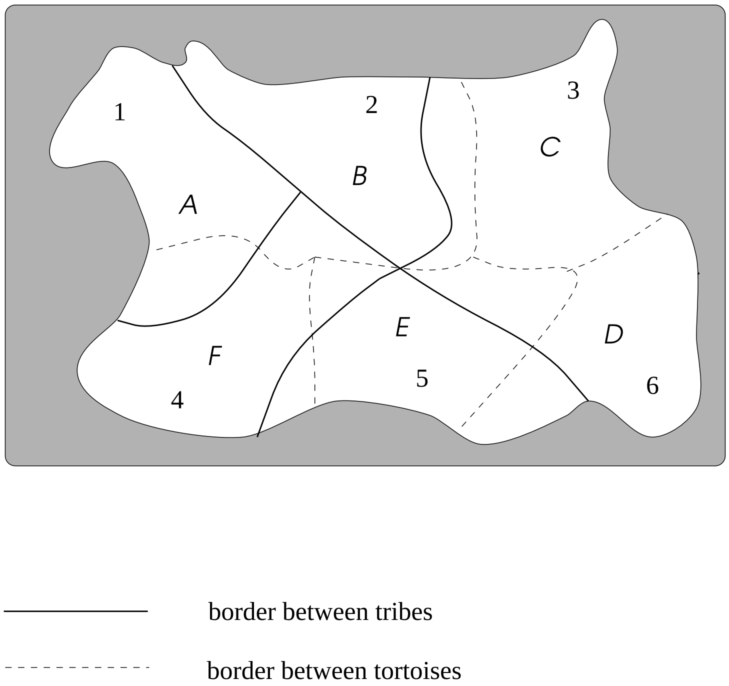

Figure 33: Six tribes and six tortoises on an island

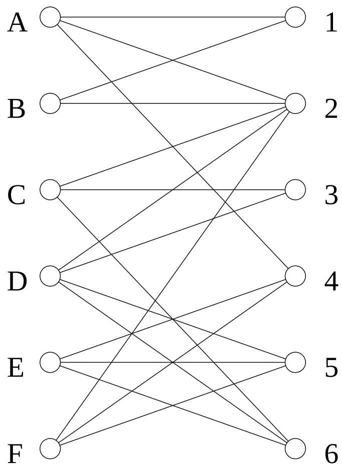

Figure 34: The graph of tribes and tortoises

So this is very similar to the problem discussed (but not solved!) in the previous section: we want to prove that a certain bipartite graph has a perfect matching. Theorem 12.1 says that for this conclusion it suffices to know that every node has the same degree. But this is too strong a condition; it is not at all fulfilled in our example (tribe A has only two tortoises to choose from, while tribe D has four).

So what property of this graph should guarantee that a perfect matching exists? Turning this question around: what would *exclude* a perfect matching?

For example, it would be bad if a tribe would not find any tortoise on its own territory. In the graph, this would correposnd to a node with degree 0. Now this is not a danger, since we know that tortoises occur everywhere on the island.

It would also be bad (as this has come up already) if two tribes could only choose one and the same tortoise. But then this tortoise would have an area of at least 200 square miles, which is not the case. A somewhat more subtle sort of trouble would arise if three tribes had only two tortoises on their combined territory. But this, too, is impossible: the two species of tortoises would cover an area of at least 300 square miles, so one of them would have to cover more than 100. More generally, we can see that the combined territory of any k tribes holds at least k species of tortoises. In terms of the graph, this means that for any k nodes on the left, there are at least k nodes on the right connected to at least one of them. We'll see in the next section that this is all we need to observe about this graph.

12.3 The main theorem

Now we state and prove a fundamental theorem about perfect matchings. This will complete the solution of the problem about tribes and tortoises, and (with some additional work) of the problem about dancing at the prom.

Theorem 12.2 (The Marriage Theorem) *A bipartite graph has a perfect matching if and only if $|A| = |B|$ and and any for subset of (say) k nodes of A there are at least k nodes in B that are connected to one of them.*

Before proving this theorem, let us discuss one more question. If we interchange "left" and "right", perfect matchings remain perfect matching. But what happens to the condition stated in the theorem? It is easy to see that it remains valid (as it should). To see this, we have to argue that if we pick any set S of k nodes in B, then they are connected to at least k nodes in B. Let $n = |A| = |B|$ and let us color the nodes in A connected to nodes in S black, the other nodes white (Figure 35). Then the white nodes are connected to at most $n-k$ nodes (since they are not connected to any node in S). Since the condition holds "from left to right", the number of white nodes is at most $n-k$. But then the number of black nodes is at most k, which proves that the condition also holds "from right to left".

Now we can turn to the proof of theorem 12.2. We shall have to refer to the condition given in the theorem so often that it will be convenient to call graphs satisfying this conditions "good" (just for the duration of this proof). Thus a bipartite graph is "good" if it has the same number of nodes left and right, and any k "left" nodes are connected to at least k "right" nodes.

It is obvious that every graph with a perfect matching is "good", so what we need to prove is the converse: *every "good" graph contains a perfect matching.* For a graph on just two nodes, being "good" means that these two nodes are connected. Thus for a graph to have

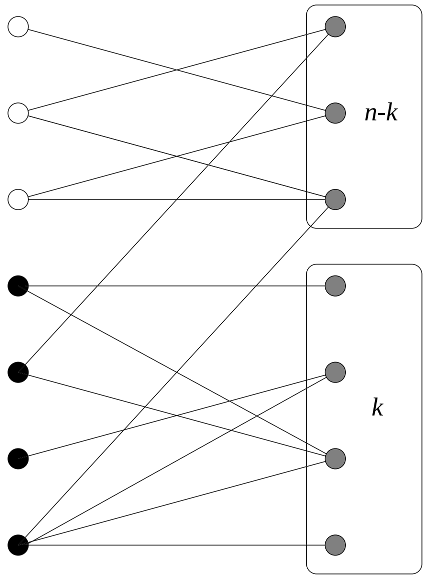

Figure 35: The good graph is also good from right to left

a perfect matching means that it can be partitioned into "good" graphs with 2 nodes. (To partition a graph means that we divide the nodes into classes, and only keep an edge between two nodes if they are in the same class.)

Now our plan is to partition our graph into two "good" parts, then partition each of these into two "good" parts etc., until we get "good" parts with 2 nodes. Then the edges that remain form a perfect matching. To carry out this plan, it suffices to prove that

if a "good" bipartite graph has more than 2 nodes, then it can be partitioned into two good bipartite graphs.

Let us try a very simple partition first: select a node $a \in A$ and $b \in B$ that are connected by an edge; let these two nodes be the first part, and the remaining nodes the other. There is no problem with the first part: it is "good". But the second part may not be good: it can have some set S of k nodes on the left connected to fewer than k nodes on the right (Figure 36). In the original graph, these k nodes were connected to at least k nodes in B; this can only be if the k-th such node was the node b. Let T denote the set of neighbors of S in the original graph. What is important to remember is that $|S| = |T|$.

Now we try another way of partitioning the graph: we take $S \cup T$ (together with the edges between them) as one part and the rest of the nodes, as the other. (This rest is not empty: the node a belongs to it, for example.)

Let's argue that both these parts are "good". Take the first graph first, take any subset of, say, j nodes in S (the left hand side of the first graph). Since the original graph was good, they are connected to at least j nodes, which are all in T by the definition of T.

For the second graph, it follows similarly that it is good, if we interchange "left" and "right". This completes the proof.

We still have to prove Theorem 12.1. This is now quite easy and is left to the reader as the following exercise.

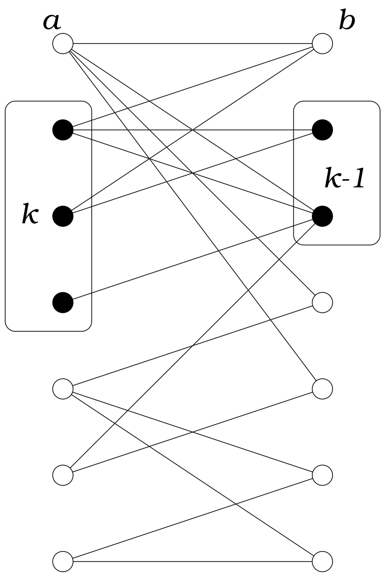

Figure 36: Goodness lost when two nodes are removed

12.4 Prove that if in a bipartite graph every node has the same degree $d \neq 0$, then the bipartite graph is "good" (and hence contains a perfect matching; this proves theorem 12.1).

12.4 How to find a perfect matching?

We have a condition for the existence of a perfect matching in a graph that is *necessary and sufficient.* Does this condition settle this issue once and forever? To be more precise: suppose that somebody gives as a bipartite graph; what is a good way to *decide* whether or not it contains a perfect matching? and how to *find* a perfect matching if there is one?

We may assume that $|A| = |B|$ (where, as before, A is the set of nodes on the left and B is the set of nodes on the right). This is easy to check, and if it fails then it is obvious that no perfect matching exists and we have nothing else to do.

One thing we can try is to look at all subsets of the edges, and see if any of these is a perfect matching. It is easy enough to do so; but there are terribly many subsets to check! Say, in our introductory example, we have 300 nodes, so $|A| = |B| = 150$; every node has degree 50, so the number of edges is $150 \cdot 50 = 7500$; the number of subsets of a set of this size is $2^{7500} > 10^{2257}$, a number that is more than astronomical...

We can do a little bit better if instead of checking all subsets of the edges, we look at all possible ways to pair up elements of A with elements of B, and check if any of these pairings matches only nodes that are connected to each other by an edge. Now the number of ways to pair up the nodes is "only" $150! \approx 10^{263}$. Still hopeless.

Can we use Theorem 12.2? To check that the necessary and sufficient condition for the existence of a perfect matching is satisfied, we have to look at every subset S of A, and see whether the number of it neighbors in B is at least as large as S itself. Since the set A has $2^{150} \approx 10^{45}$ subsets, this takes a much smaller number of cases to check than any of the previous possibilities, but still astronomical!

So theorem 12.2 does not really help too much in deciding whether a given graph has a perfect matching. We have seen that it does help in *proving* that certain properties of a graph imply that the graph has a perfect matching. We'll come back to this theorem later and discuss its significance. Right now, we have to find some other way to deal with our problem.

Let us introduce one more expression: by a *matching* we mean a set of edges that have no endpoint in common. A perfect matching is the special case when, in addition, the edges cover all the nodes. But a matching can be much smaller: the empty set, or any edge in itself, is a matching.

Let's try to construct a perfect matching in our graph by starting with the empty set and building up a matching one by one. So we select two nodes that are connected, and mark the edge between them; then we select two other nodes that are connected, and mark the edge between them etc. we can do this until no two unmatched nodes are connected by an edge. The edges we have marked form a matching M. If we are lucky, then M is a perfect matching, and we have nothing else to do. But what do we do if M is not perfect? Can we conclude that the graph has no perfect matching at all? No, we cannot; it may happen that the graph has a perfect matching, but we made some unlucky choices when selecting the edges of M.

12.5 Show by an example that it may happen that a bipartite graph G has a perfect matching but, if we are unlucky, the matching M constructed above is not perfect.

12.6 Prove that if G has a perfect matching, then M mathes up at least half of the nodes.

So suppose that we have constructed a matching M that is not perfect. We have to try to increase its size by "backtracking", i.e., by deleting some of its edges and replacing them by more edges. But how to find the edges we want to replace?

The trick is the following. We look for a path P in G of the following type: P starts and ends at nodes u and v that are unmatched by M; and every second edge of P belongs to M (Figure 37). Such a path is called an *augmenting path*. It is clear that an augmenting path P contains an odd number of edges, and in fact the number of its edges not in M is one larger than the number of its edges in M.

If we find an augmenting path P, we can delete those edges of P that are in M and replace them by those edges of P that are not in M. It is clear that this results in a matching M' that is larger than M by one edge. (The fact that M' is a matching follows from the observation that the remaining edges of M cannot contain any node of P: the two endpoints of P were supposed to be unmatched, while the interior nodes of P were matched by edges of M that we deleted.) So we can repeat this until we either get a perfect matching, or a matching M for which no augmenting path exists.

So we have two questions to answer: how to find an augmenting path, if it exists? and if it does not exist, does this mean that there is no perfect matching at all? It will turn out that an answer to the first question will also imply the (affirmative) answer to the second.

Let U be the set of unmatched nodes in A and let W be the set of unmatched nodes in B. As we noted, any augmenting path must have an odd number of edges and hence it must connect a node in U to a node in W. Let us try to find such an augmenting path starting from some node in U. Let's say that a path Q is *almost augmenting* if it starts at a node in U, ends at a node in A, and every second edge of it belonds to M. An almost augmenting path must have an even number of edges, and must end with an edge of M.

What we want to do is to find the set of nodes in A that can be reached on an almost augmenting path. Let's agree that we consider a node in U as an almost augmenting path in itself (of length 0); then we know that every node in U has this property. Starting with

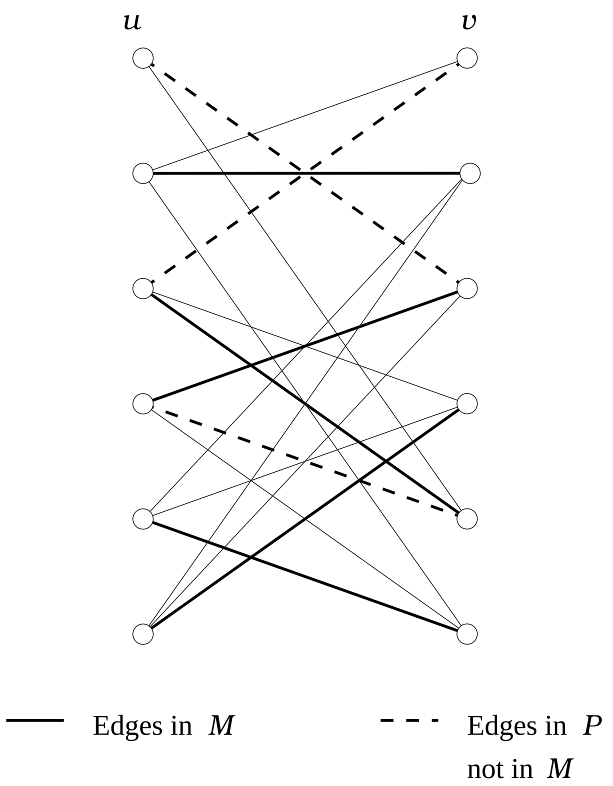

Figure 37: An augmenting path in a bipartite graph

$S = U$, we build up a set S gradually. At any stage, the set S will consist of nodes we already know are reachable by some almost augmenting path. We denote by T the set of nodes in B that are matched with nodes in S (Figure 38). Since the nodes of U have nothing matched with them and they are all in S, we have

$$|S| = |T| + |U|.$$

We look for an edge that connects a node $s \in S$ to some node $r \in B$ that is *not* in T. Let Q be an almost augmenting path starting at some node $u \in U$ and ending at s. Now there are two cases to consider:

— If r is unmatched (which means that it belongs to W) then appending the edge sr to Q we get an augmenting path P. So we can increase the size of M (and forget about S and T).

— If r is matched with a node $q \in A$, then we can append the edges sr and rq to Q, to get an almost augmenting path from u to q. So we can add u to S.

So if we find an edge connecting a node in S to a node not in T, we can either increase the size of M or the increase the set S (and leave M as it was). Sooner of later we must encounter a situation where either M is a perfect matching (and we are done), or M is not perfect but no edge connects S to any node outside T.

So what to do in this case? Nothing! If this occurs, we can conclude that there is no perfect matching at all. In fact, all neighbors of the set S are in T, and $|T| = |S| - |U| < |S|$. We know that this implies that there is no perfect matching at all in the graph.

Figure 39 shows how this algorithm finds a matching in the bipartite graph that is a subgraph if the "grid".

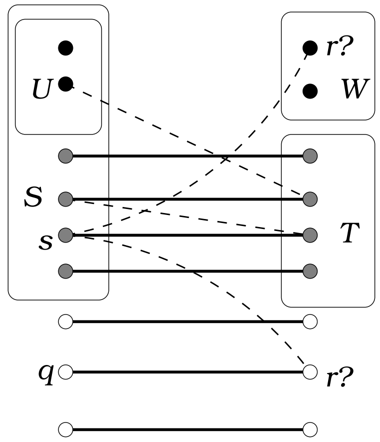

Figure 38: Reaching nodes by almost augmenting paths. Only edges on these paths, and of M, are shown.

To sum up, we do the following. At any point in time, we'll have a matching M and a set S of nodes in A that we know can be reached on almost augmenting paths. If we find an edge connecting S to a node not matched with any node in S, we can either increase the size of M or the set S, and repeat. If no such edge exists, either M is perfect or no perfect matching exists at all.

Remark. In this chapter, we restricted our attention to matchings in bipartite graphs. One can, of course, define matchings in general (nonbipartite) graphs. It turns out that both the necessary and sufficient condition given in theorem 12.2 and the algorithm described in this section can be extended to non-bipartite graphs. This requires, however, quite a bit more involved methods, which lie beyond the scope of this book.

12.7 Follow how the algorithm works on the graph in Figure 40.

12.8 Show how the description of algorithm above contains a new proof of theorem 12.2.

12.5 Hamiltonian cycles

A Hamiltonian cycle is a cycle that contains all nodes of a graph. We have met a similar notion before: tavelling salesman tours. Indded, travelling salesman tours can be viewed as Hamiltonian cycles in the complete graph on the given set of nodes.

Hamiltonian cycles are quite similar to matchings: in a perfect matching, every node belongs to exactly one edge; in a Hamiltonian cycle, every node belongs to exactly two edges. But much less is known about them than about matchings. For example, no efficient way is known to check whether a given graph has a Hamiltonian cycle (even if it is bipartite), and no useful necessary and sufficient condition for the existence of a Hamiltonian cycle is known. If you solve exercise 12.5, you'll get a feeling about the difficulty of the Hamiltonian cycle problem.

12.9 Decide whether or not the graphs in Figure 41 have a Hamiltonian cycle.

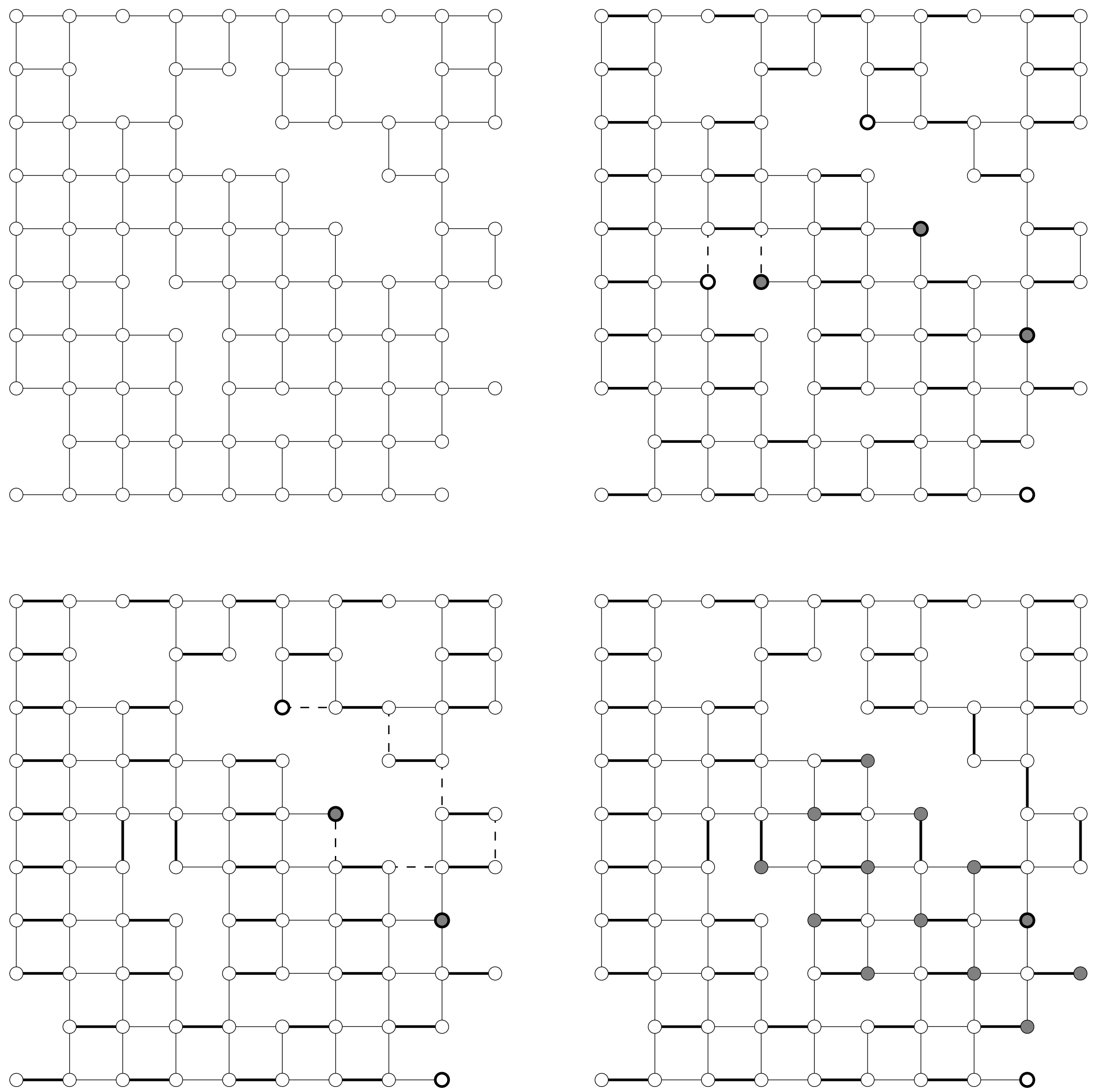

Figure 39: A graph for trying out the algorithm

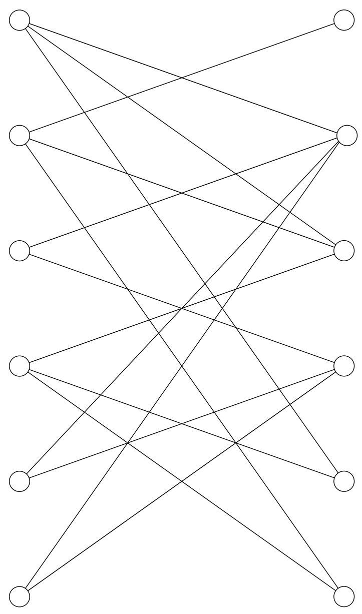

Figure 40: A graph for trying out the algorithm

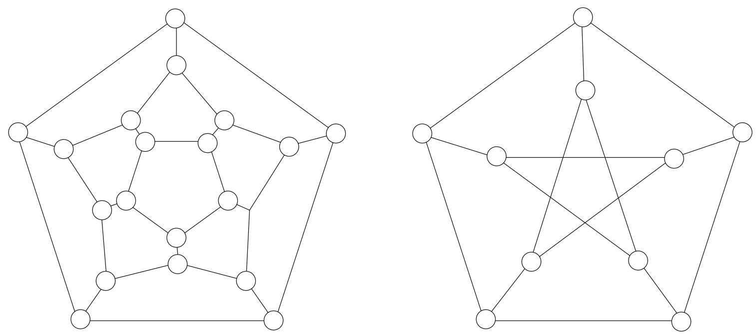

Figure 41: Do these graphs have a Hamilton cycle?

13 A Connecticut class in King Arthur's court

In the court of King Arthur there dwelt 150 knights and 150 ladies-in-waiting. The king decided to marry them off, but the trouble was that some pairs hated each other so much that they would not even get married, let alone speak! King Arthur tried several times to pair them off but each time he ran into conflicts. So he summoned Merlin the Wizard and ordered him to find a pairing in which every pair was willing to marry. Now Merlin had supernatural powers and he saw immediately that none of the 150! possible pairings was feasible, and this he told the king. But Merlin was not only a great wizard, but a suspicious character as well, and King Arthur did not quite trust him. "Find a pairing or I shall sentence you to be imprisoned in a cave forever!" said Arthur.

Fortunately for Merlin, he could use his supernatural powers to browse forthcoming scientific literature, and he found several papers in the early 20th century that gave the *reason* why such a pairing could not exist. He went back to the King when all the knights and ladies were present, and asked a certain 56 ladies to stand on one side of the king and 95 knights on the other side, and asked: "Is any one of you ladies, willing to marry any of these knights?", and when all said "No!", Merlin said: "O King, how can you command me to find a husband for each of these 56 ladies among the remaining 55 knights?" So the king, whose courtly education did include the pigeon-hole principle, saw that in this case Merlin had spoken the truth and he graciously dismissed him.

Some time elapsed and the king noticed that at the dinners served for the 150 knights at the famous round table, neighbors often quarrelled and even fought. Arthur found this bad for the digestion and so once again he summoned Merlin and ordered him to find a way to seat the 150 knights around the table so that each of them should sit between two friends. Again, using his supernatural powers Merlin saw immediately that none of the 150! seatings would do, and this he reported to the king. Again, the king bade him find one or explain why it was impossible. "Oh I wish there were some simple reason I could give to you! With some luck there could be a knight having only one friend, and so you too could see immediately that what you demand from me is impossible. But alas!, there is no such simple reason here, and I cannot explain to you mortals why no such seating exists, unless you are ready to spend the rest of your life listening to my arguments!" The king was naturally unwilling to do that and so Merlin has lived imprisoned in a cave ever since. (A severe loss for applied mathematics!)

The moral of this tale is that there are properties of graphs which, when they hold, are easily proven to hold. If a graph has a perfect matching, or a Hamilton cycle, this can be "proved"easily by exhibiting one. If a bipartite graph does *not* have a perfect matching, then this can be "proved" by exhibiting a subset X of one color class which has fewer than $|X|$ neighbors in the other. The reader (and King Arthur!) are directed to Figure 42 in which graph G has a perfect matching (indicated by the heavy lines), but graph H does not. To see the latter, consider the four black points and their neighbors.

Most graph-theoretic properties which interest us have this logical structure. Such a property is called (in the jargon of computer science) and NP-property (if you really want to knwo, NP is and abbreviation of *Nondeterministic Polynomial Time*, but it would be difficult to explain where this highly technical phrase comes from). The two problems that Merlin had to face — the existence of a perfect matching and the existence of a Hamilton cycle — are clearly NP-properties. But NP-properties also appear quite frequently in other parts of mathematics. A very important NP-property of natural numbers is their *compositeness*: if a natural number is composite then this can be exhibited easily by showing a decomposition $n = ab$ $(a, b > 1)$.

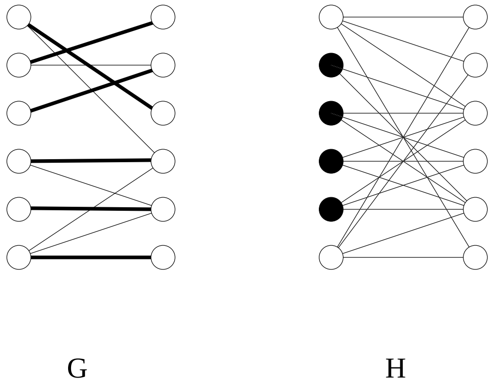

Figure 42: A bigraph with a perfect matching and one without

The remarks we have made so far explain how Merlin can remain free if he is lucky and the task assigned to him by King Arthur has a solution. For instance, suppose he could find a good way to seat the knights for dinner. He could then convince King Arthur that his seating plan was "good" by asking if there was anybody sitting beside and enemy of his. This shows that the property of the corresponding "friendship graph" that it contains a Hamilton cycle is an NP-property. But how could he survive Arthur's wrath in the case of the marriage problem and not in the case of the seating problem when these problems do *not* have solutions? What distinguishes the non-existence of a Hamilton cycle in a graph from the non-existence of a perfect matching in a bigraph? From our tale, we hope the answer is clear: *the non-existence of a perfect matching in a bigraph is also an NP-property* (this is a main implication of Frobenius' Theorem), while the non-existence of a Hamilton cycle in a graph is not! (To be precise, no proof of this latter fact is known, but there is strong evidence in favor of it.

So for certain NP-properties the negation of the property is again an NP-property. A theorem asserting the equivalence of an NP-property with the negation of another NP-property is called a *good characterization*. There are famous good characterizations throughout graph theory and elsewhere.

Many NP-properties are even better. Facing the problem of marrying off his knights and ladies, Arthur himself (say, after attending this class) could decide himself whether or not it was solvable: he could run the algorithm described in section 12.4. A lot of work, but probably doable with the help of quite ordinary people, without using the supernatural talents of Merlin. Properties which can be decided efficiently are called properties *in the class P* (here P stand for *Polynomial Time*, an exact but quite technical definition of the phrase "efficiently"). Many other simple properties of graphs discussed in this book also belong to the class: connectivity, or the existence of a cycle.

Answers to exercises

2 Let us count!

2.1 A party

2.1. 7!.

2.1. Carl: $15 \cdot 2^3 = 120$. Diane: $15 \cdot 3! = 90$.

2.2 Sets

2.2. (a) all houses in a street; (b) an olympic team; (c) class of '99; (d) all trees in a forest; (e) the set of rational numbers; (f) a circle in the plane.

2.2. (a) soldiers; (b) people; (c) books; (d) animals.

2.2. (a) all cards in a deck; (b) all spades in a deck; (c) a deck of Swiss cards; (d) non-negative integers with at most two digits; (e) non-negative integers with exactly two digits; (f) inhabitants of Budapest, Hungary.

2.2. Alice, and the set whose only element is the number 1.

2.2. $6 \cdot 5/2 = 15$.

2.2. No.

2.2. $\emptyset, \{0\}, \{1\}, \{3\}, \{0,1\}, \{0,3\}, \{1,3\}, \{0,1,3\}$. 8 subsets.

2.2. women; people at the party; students of Yale.

2.2. $\mathbb{Z}$ or $\mathbb{Z}_+$. The smallest is $\{0,1,3,4,5\}$.

2.2. (a) $\{a,b,c,d,e\}$. (b) The union operation is associative. (c) The union of any set of sets consists of those elements which are lements of at least one of the sets.

2.2. The union of a set of sets $\{A_1, A_2, \ldots, A_k\}$ is the smallest set containing each A_i as a subset.

2.2. $6, 9, 10, 14$.

2.2. The cardinality of the union is at least the larger of n and m and at most $n+m$.

2.2. (a) $\{1,3\}$; (b) $\emptyset$; (c) $\{2\}$.

2.2. The cardinality of the intersection is at most the minimum of n and m.

2.2. The common elements of A and B are counted twice on both sides; the elements in either A or B but not both are counted once on both sides.

2.2 (a) The set of negative even integers and positive odd integers. (b) B.

2.3 The number of subsets

2.3. Powers of 2.

2.3. 2^{n-1}.

2.3. (a) $2 \cdot 10^n - 1$; (b) $2 \cdot (10^n - 10^{n-1}$.

2.3. 101.

2.3 $1 + \lfloor n \lg 2 \rfloor$.

2.4 Sequences

2.4. The trees have 9 and 8 leaves, respectively.

2.4. $5 \cdot 4 \cdot 3 = 60$.

2.4. 3^{13}.

2.4. $6 \cdot 6 = 36$.

2.4. 12^{20}.

2.4. $(2^{20})^{12}$.

2.5 Permutations

2.5. $n!$.

2.5. (b) $7 \cdot 5 \cdot 3 = 105$. In general, $(2n-1) \cdot (2n-3) \cdot \ldots \cdot 3 \cdot 1$.

2.5. (a) $n(n-1)/2$ is larger for $n \geq 4$. (b) 2^n is larger for $n \geq 5$.

2.5. (a) This is true for $n \geq 10$. (b) $2^n/n^2 > n$ for $n \geq 10$.

3 Induction

3.1 The sum of odd numbers

3.1. One of n and $n+1$ is even, so the product $n(n+1)$ is even. By induction: true for $n=1$; if $n > 1$ then $n(n+1) = (n-1)n + 2n$, and $n(n-1)$ is even by the induction hypothesis, $2n$ is even, and the sum of two even numbers is even.

3.1. True for $n = 1$. If $n > 1$ then

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

3.1. The youngest person will count $n-1$ handshakes. The 7-th oldest will count 6 handshakes. So they count $1 + 2 + \ldots + (n-1)$ handshakes. We also know that there are $n(n-1)/2$ handshakes.

3.1. Compute the area of the rectangle in two different ways.

3.1. By induction on n true for $n = 1$. For $n > 1$, we have

$$1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 + \ldots + (n-1) \cdot n = \frac{(n-2) \cdot (n-1) \cdot n}{3} + (n-1) \cdot n = \frac{(n-1) \cdot n \cdot (n+1)}{3}.$$

3.1. If n is even, then $1 + n = 2 + (n-1) = ... = (\frac{n}{2} - 1) + \frac{n}{2} = n + 1$, so the sum is $\frac{\frac{n}{2}(n+1) = n(n+1)}{2}$. If n is odd then we have to add the middle term separately.

3.1. If n is even, then $1 + (2n-1) = 3 + (2n-3) = \ldots = (n-1) + (n+1) = 2n$, so the sum is $\frac{n}{2}(2n) = n^2$. Again, is n is odd the solution is similar, but we have to add the middle term separately.

3.1. By induction. True for $n = 1$. If $n > 1$ then

$$1^2 + 2^2 + \ldots + (n-1)^2 = (1^2 + 2^2 + \ldots + (n-1)^2) + n^2 = \frac{(n-1)n(2n-1)}{6} + n^2 = \frac{n(n+1)(2n+1)}{6}.$$

3.1. By induction. True for $n = 1$. If $n > 1$ then

$$2^0 + 2^1 + 2^2 + \ldots + 2^{n-1} = (2^0 + 2^1 + \ldots + 2^{n-2}) + 2^{n-1} = (2^{n-1} - 1) + 2^{n-1} = 2^n - 1.$$

3.2 Subset counting revisited

3.2. (Strings) True for $n = 1$. If $n > 1$ then to get a string of length n we can start with a string of length $n - 1$ (this can be chosen in k^{n-1} ways by the induction hypothesis) and append an element (this can be chosen in k ways). So we get $k^{n-1} \cdot k = k^n$.

(Permutations) True for $n = 1$. To seat n people, we can start with seating the oldest (this can be done in n ways) and then seating the rest (this can be done in $(n-1)!$ ways be the induction hypothesis). We get $n \cdot (n-1)! = n!$.

3.2. True if $n = 1$. Let $n > 1$. The number of handshakes between n people is the number of handshakes by the oldest person $(n-1)$ plus the number of handshakes between the remaining $n-1$ (which is $(n-1)(n-2)/2$ by the induction hypothesis). We get $(n-1)+(n-1)(n-2)/2 = n(n-1)/2$.

3.3. By induction. True if $n = 1$. Let $n > 1$. Assume the description of the coloring is valid for the first $n-1$ circles. If we add the n-th, the color and the parity don't change outside this circle; both change inside the circle. So the description remains valid.

3.3. (a) By induction. True for 1 line. Adding a line, we recolor all regions on one side.

(b) One possible description: designate a direction as "up". Let p any point not on any of the lines. Start a semiline "up" from P. Count how many of the given lines intersect it. Color according to the parity of this intersection number.

3.3. We did not check the base case $n = 1$.

3.3. The proof uses that there are at least four lines. But we only checked $n = 1, 2$ as base cases. The assertion is false for $n = 3$ and also for every value after that.

3.4 Counting regions

3.4. True for $n = 1$. Let $n > 1$. Delete any line. The remaining lines divide the plane into $(n-1)n/2 + 1$ regions by the induction hypothesis. The last line cuts n of these into two. So we get

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

4 Counting subsets

4.1 The number of ordered subsets

4.1. (I don't think you could really draw the whole tree; it has almost 10^20 leaves. It has 11 levels of nodes.)

4.1. (a) 100!. (b) 90!. (c) $100!/90! = 100 \cdot 99 \cdot \ldots \cdot 91$.

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

4.1. In one case, repetition is not allowed, while in the other case, it is allowed.

4.2 The number of subsets of a given size

4.2. Handshakes; lottery; hands in bridge.

4.2. See next chapter.

4.2.

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

4.2. Solution of (b) ((a) is a special case)). The identity is

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}.$$

The right hand side counts k-subsets of an n-element set by separately counting those that do not contain a given element and those that do.

4.2. The number of k-element subsets is the same as the number of $(n-k)$-element subsets, since the complement of a k-subset is an $(n-k)$-subset and vice versa.

4.2. Both sides count all subsets of an n-element set.

4.2. Both sides count the number of ways to divide an a-element set into three sets with $a-b$, $b-c$, and c elements.

4.3 The Binomial Theorem

4.3.

$$\begin{aligned}
(x+y)^n &= (x+y)^{n-1}(x+y) \\
&= \left(x^{n-1} + \binom{n-1}{1}x^{n-2}y + \dots \right. \\
&\quad \left. + \binom{n-1}{n-2}xy^{n-2} + \binom{n-1}{n-1}y^{n-1}\right)(x+y) \\
&= x^{n-1}(x+y) + \binom{n-1}{1}x^{n-2}y(x+y) + \dots \\
&\quad + \binom{n-1}{n-2}xy^{n-2}(x+y) + \binom{n-1}{n-1}y^{n-1}(x+y) \\
&= (x^n + x^{n-1}y) + \binom{n-1}{1}(x^{n-1}y + x^{n-2}y^2) + \dots + \binom{n-1}{n-2}(x^2y^{n-2} + xy^{n-1}) \\
&\quad + \binom{n-1}{n-1}(xy^{n-1} + y^n) \\
&= x^n + \left(1 + \binom{n-1}{1}\right)(x^{n-1}y + x^{n-2}y) + \dots \\
&\quad + \left(\binom{n-1}{n-2} + \binom{n-1}{n-1}\right)xy^{n-1} + y^n \\
&= x^n + \binom{n}{1}x^{n-1}y + \binom{n}{2}x^{n-2}y^2 + \dots + \binom{n}{n-1}xy^{n-1} + y^n.
\end{aligned}$$

4.3. (a) $(1-1)^n = 0$. (b) By $\binom{n}{k} = \binom{n}{n-k}$.

4.3. Both sides count all subsetsof an n-element set.

4.4 Distributing presents

4.4.

$$\binom{n}{n_1} \cdot \binom{n-n_1}{n_2} \cdot \ldots \cdot \binom{n}{n_k}$$

$$= \frac{n!}{n_1!(n-n_1)!}\frac{(n-n_1)!}{n_2!(n-n_1-n_2)!}\cdots\frac{(n-n_1-\ldots-n_{k-2})!}{n_{k-1}!(n-n_1-\ldots-n_{k-1})!} = \frac{n!}{n_1!n_2!\ldots n_k!},$$

since $n - n_1 - \ldots - n_{k-1} = n_k$.

4.4. (a) $n!$ (distribute positions instead of presents). (b) $n(n-1)\ldots(n-k+1)$ (distribute as "presents" the first k positions at the competition and $n-k$ participation certificates). (c) $\binom{n}{n_1}$. (d) Chess seating in Diane's sense (distribute players to boards).

4.4. (a) $[n = 8]$ $8!$. (b) $8! \cdot \binom{8}{4}$. (c) $(8!)^2$.

4.5 Anagrams

4.5. $13!/2^3$.

4.5. COMBINATORICS.

4.5. Most: any word with 13 different letters; least: any word with 13 identical letters.

4.5. (a) 26^6.

(b) $\binom{26}{4}$ ways to select the four letters that occur; for each selection, $\binom{4}{2}$ ways to select the two letters that occur twice; for each selection, we distribute 6 positions to these letters (2 of them get 2 positions), this gives $\frac{6!}{2!2!}$ ways. Thus we get $\binom{26}{4}\binom{4}{2}\frac{6!}{2!2!}$. (There are many other ways to arrive at the same number!)

(c) Number of ways to partition 6 into the sum of positive integers:

$$6 = 6 = 5+1 = 4+2 = 4+1+1 = 3+3 = 3+2+1 = 3+1+1+1 = 2+2+2$$

$$= 2+2+1+1 = 2+1+1+1+1 = 1+1+1+1+1+1$$

which makes 11 possibities.

(d) This is too difficult in this form. What I meant is the following: how many words of length n are there such that none is an anagram of another? This means distributing n pennies to 26 children, and so the answer is $\binom{n+25}{26}$.

4.6 Distributing money

4.6. $\binom{n-k-1}{k-1}$.

4.6. $\binom{n+k-1}{\ell+k-1}$.

4.6. $\binom{kp+k-1}{k-1}$.

5 Pascal's Triangle

5. This is the same as $\binom{n}{k} = \binom{n}{n-k}$.

5. $\binom{n}{0} = \binom{n}{n} = 1$ (e.g. by the general formula for the binomial coefficients).

5.1.

$$1 + \binom{n}{1} + \binom{n}{2} + \ldots + \binom{n}{n-1} + \binom{n}{n}$$

$$= 1 + \left(\binom{n-1}{0} + \binom{n-1}{1}\right) + \left(\binom{n-1}{1} + \binom{n-1}{2}\right) + \ldots + \left(\binom{n-1}{n-2} + \binom{n-1}{n-1}\right) + 1$$

$$= 2\left(\binom{n-1}{0} + \binom{n-1}{1} + \ldots + \binom{n-1}{n-2} + \binom{n-1}{n-1}\right) = 2 \cdot 2^{n-1} = 2^n.$$

5.1 Identities in the Pascal Triangle

5.1.

$$1+\binom{n}{1}+\binom{n}{2}+\ldots+\binom{n}{n-1}+\binom{n}{n}$$

$$=1+\left(\binom{n-1}{0}+\binom{n-1}{1}\right)+\left(\binom{n-1}{1}+\binom{n-1}{2}\right)+\ldots+\left(\binom{n-1}{n-2}+\binom{n-1}{n-1}\right)+1$$

$$=2\left(\binom{n-1}{0}+\binom{n-1}{1}+\ldots+\binom{n-1}{n-2}+\binom{n-1}{n-1}\right)=2\cdot 2^{n-1}=2^n.$$

5.1. The coefficient of $x^n y^n$ in

$$\left(\binom{n}{0}x^n+\binom{n}{1}(x^{n-1}y+x^{n-2}y)+\ldots+\binom{n}{n-1}xy^{n-1}+\binom{n}{n}y^n\right)^2$$

is

$$\binom{n}{0}\binom{n}{n}+\binom{n}{1}\binom{n}{n-1}+\ldots+\binom{n}{n-1}\binom{n}{1}+\binom{n}{n}\binom{n}{0}.$$

5.1. The left hand side counts all k-element subsets of an $(n+m)$-element set by distinguishing them according to how many elements they pick up from the first n.

5.1. If the largest element is j, the rest can be chosen $\binom{j-1}{k}$ ways.

5.2 A bird's eye view at the Pascal Triangle

5.2. $n = 3k+2$.

5.2. $k = \lfloor (n-\sqrt{2n})/2 \rfloor$. (This is not easy: one looks at the difference of differences:

$$\left(\binom{n}{k+1}-\binom{n}{k}\right)-\left(\binom{n}{k}-\binom{n}{k-1}\right),$$

and determines the value of k where it turns negative.)

5.2. (a) 2^n is a sum with positive terms in which $\binom{n}{4}$ is only one of the terms.

(b) Assume that $n > 200$. Then

$$\frac{2^n}{n^3} \ge \frac{\binom{n}{4}}{n^3} = \frac{(n-1)(n-2)(n-3)}{24n^2} > \frac{(n/2)^3}{24n^2} = \frac{n}{192} > 1.$$

5.2.

$$\binom{n}{n/2} = \frac{n!}{((n/2)!)^2} \sim \frac{\left(\frac{n}{e}\right)^n\sqrt{2\pi n}}{\left(\left(\frac{n}{2e}\right)^(n/2)\sqrt{\pi n}\right)^2} = \frac{2^n\sqrt{2}}{\sqrt{\pi n}}.$$

5.2. Using

$$\binom{2m}{m}\Big/\binom{2m}{m-t} > \frac{t^2}{m},$$

it is enough to find a $t > 0$ for which $t^2/m \ge 1/c$. Solving for t, we get that $t = \lceil \sqrt{m/c} \rceil$ is a good choice.

5. (a): see (c).
(b) We prove by induction on s that for $0 \le s \le m-t$,

$$\binom{2m}{m-s} \Big/ \binom{2m}{m-t-s} > \frac{t^2}{m}.$$

For $s=0$ this is just the theorem we already know. Let $s>0$, then then

$$\binom{2m}{m-s} = \frac{m-s+1}{m+s}\binom{2m}{m-s+1}$$

and

$$\binom{2m}{m-t-s} = \frac{m-s-t+1}{m+s+t}\binom{2m}{m-s-t+1}.$$

Hence

$$\binom{2m}{m-s} \Big/ \binom{2m}{m-t-s} = \frac{(m-s+1)(m+s+t)}{(m+s)(m-s-t+1)}\left(\binom{2m}{m-s+1} \Big/ \binom{2m}{m-t-s+1}\right).$$

Since

$$\frac{(m-s+1)(m+s+t)}{(m+s)(m-s-t+1)} > 1,$$

it follows that

$$\binom{2m}{m-s} \Big/ \binom{2m}{m-t-s} > \binom{2m}{m-s+1} \Big/ \binom{2m}{m-t-s+1} > \frac{t^2}{m}$$

by the induction hypothesis.
(c)

$$\begin{aligned}
\frac{\binom{2m}{m-s}}{\binom{2m}{m-t-s}} = \frac{\frac{(2m)!}{(m-s)!(m+s)!}}{\frac{(2m)!}{(m-t-s)!(m+t+s)!}} &= \frac{(m+t+s)(m+t+s-1)\dots(m+s+1)}{(m-s)(m-s-1)\dots(m-s-t+1)} \\
&= \frac{m+t+s}{m-s}\cdot\frac{m+t+s-1}{m-s-1}\cdot\ldots\cdot\frac{m+s+1}{m-t-s+1} \\
&= \left(1+\frac{t+2s}{m-s}\right)\cdot\left(1+\frac{t+2s}{m-s-1}\right)\cdot\ldots\cdot\left(1+\frac{t}{m-s-t+1}\right) \\
&\ge \left(1+\frac{t+2s}{m}\right)^t > 1+t\frac{t+2s}{m}.
\end{aligned}$$

6 Combinatorial probability

6.1 Events and probabilities

6.1. The union of two events A and B corresponds to "A or B".
6.1. It is the sum of some of the probabilities of outcomes, and even if add all we get just 1.
6.1. $\mathsf{P}(E) = \frac{1}{2}$, $\mathsf{P}(T) = \frac{1}{3}$.
6.1. The same probabilities $\mathsf{P}(s)$ are added up on both sides.

6.1. Every probability $\mathsf{P}(s)$ with $s \in A \cap B$ is added twice to sides; every proability $\mathsf{P}(s)$ with $s \in A \cup B$ but $s \notin A \cap B$ is added once to both sides.

6.2 Independent repetition of an experiment

6.2. The pairs $(E, T), (O, T), (L, T)$ are independent. The pair (E, O) is exclusive.

6.2. $\mathsf{P}(\emptyset \cap A) = \mathsf{P}(emptyset) = 0 = \mathsf{P}(emptyset)\mathsf{P}(A)$. The set S also has this property: $\mathsf{P}(S \cap A) = \mathsf{P}(A) = \mathsf{P}(S)\mathsf{P}(A)$.

6.2. $\mathsf{P}(A) = \frac{|S|^{n-1}}{|S|^n} = \frac{1}{|S|}$, $\mathsf{P}(B) = \frac{|S|^{n-1}}{|S|^n} = \frac{1}{|S|}$, $\mathsf{P}(A \cap B) = \frac{|S|^{n-2}}{|S|^n} = \frac{1}{|S|^2} = \mathsf{P}(A)\mathsf{P}(B)$.

7 Fibonacci numbers

7.1 Fibonacci's exercise

7.1. Because we use the two previous elements to compute the next.

7.1. F_{n+1}.

7.2. It is clear from the recurrence that two odd members are followed by an even, then by two odd.

7.2. We formulate the following nasty looking statement: *if n is divisible by 5, then so is F_n; if n has remainder 1 when divided by 5, then F_n has remainder 1; if n has remainder 2 when divided by 5, then F_n has remainder 1; if n has remainder 3 when divided by 5, then F_n has remainder 2; if n has remainder 4 when divided by 5, then F_n has remainder 3.* This is then easily proved by induction on n.

7.2. By induction. All of them are true for $n = 1$ and $n = 2$. Assume that $n \geq 3$.

(a) $F_1 + F_3 + F_5 + \ldots + F_{2n-1} = (F_1 + F_3 + \ldots + F_{2n-3}) + F_{2n-1} = F_{2n-2} + F_{2n-1} = F_{2n}$.

(b) $F_0 - F_1 + F_2 - F_3 + \ldots - F_{2n-1} + F_{2n}(F_0 - F_1 + F_2 - \ldots + F_{2n-2}) - (F_{2n-1} + F_{2n}) = (F_{2n-3} - 1) + F_{2n-2} = F_{2n-1} - 1$.

(c) $F_0^2 + F_1^2 + F_2^2 + \ldots + F_n^2(F_0^2 + F_1^2 + \ldots + F_{n-1}^2) + F_n^2 = F_{n-1}F_n + F_n^2 = F_n(F_{n-1} + F_n) == F_n \cdot F_{n+1}$.

(d) $F_{n-1}F_{n+1} - F_n^2 = F_{n-1}(F_{n-1} + F_n) - F_n^2 = F_{n-1}^2 + F_n(F_{n-1} - F_n) = F_{n-1}^2 - F_nF_{n-2} = -(-1)^{n-1} = (-1)^n$.

7.2. The identity is

$$\binom{n}{0} + \binom{n-1}{1} + \binom{n-2}{2} + \ldots + \binom{n-k}{k} = F_{n+1},$$

where $k = \lfloor n/2 \rfloor$. Proof by induction. True for $n = 0$ and $n = 1$. Let $n \geq 2$. Assume that n is odd; the even case is similar, just the last term below needs a little different treatment.

$$\binom{n}{0} + \binom{n-1}{1} + \binom{n-2}{2} + \ldots + \binom{n-k}{k}$$

$$= 1 + \left(\binom{n-2}{0} + \binom{n-2}{1}\right) + \left(\binom{n-3}{1} + \binom{n-3}{2}\right) + \ldots + \left(\binom{n-k-1}{k-1} + \binom{n-k-1}{k}\right)$$

$$= \left(\binom{n-1}{0} + \binom{n-2}{1} + \binom{n-3}{2} + \ldots + \binom{n-k-1}{k}\right)$$

$$+\left(\binom{n-2}{0}+\binom{n-3}{1}+\ldots+\binom{n-k-1}{k-1}\right)=F_n+F_{n-1}=F_{n+1}.$$

7.2. The "diagonal" is in fact a very long and narrow paralellogram with area 1. The trick depends on the fact $F_{n+1}F_{n-1}-F_n^2=(-1)^n$ is very small compared to F_n^2.

7.3 A formula for the Fibonacci numbers

7.3. True for $n=0,1$. Let $n\geq 2$. Then by the induction hypothesis,

$$F_n=F_{n-1}+F_{n-2}$$

$$=\frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^{n-1}-(\left(\frac{1-\sqrt{5}}{2}\right)^{n-1}\right).+\frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^{n-2}-(\left(\frac{1-\sqrt{5}}{2}\right)^{n-2}\right).$$

$$=\frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{n-2}\left(\frac{1+\sqrt{5}}{2}+1\right)+\left(\frac{1-\sqrt{5}}{2}\right)^{n-2}\left(\frac{1-\sqrt{5}}{2}+1\right)\right]$$

$$=\frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^{n}-(\left(\frac{1-\sqrt{5}}{2}\right)^{n}\right).$$

7.3. For $n=0$ and $n=1$, if we require that H_n is of the given form we get

$$H_0=1=a+b,\qquad H_1=3=a\frac{1+\sqrt{5}}{2}+b\frac{1-\sqrt{5}}{2}.$$

Solving for a and b, we get

$$a=\frac{1+\sqrt{5}}{2},b=\frac{1-\sqrt{5}}{2}.$$

Then

$$H_n=\left(\frac{1+\sqrt{5}}{2}\right)^{n+1}+\left(\frac{1-\sqrt{5}}{2}\right)^{n+1}$$

follows by induction on n just like in the previous problem.

7.3.

$$I_n=\frac{1}{2\sqrt{5}}\left((2+\sqrt{5})^n-(2-\sqrt{5})^n\right).$$

8 Integers, divisors, and primes

8.1 Divisibility of integers

8.1. $a=a\cdot 1=(-a)\cdot(-1)$.

8.1. (a) even; (b) odd; (c) $a=0$.

8.1. (a) If $b=am$ and $c=bn$ then $c=amn$. (b) If $b=am$ and $c=an$ then $b+c=a(m+n)$ and $b-c=a(m-n)$. (c) If $b=am$ and $a,b>0$ then $m>0$, hence $m\geq 1$ and so $b\geq a$. (d) Trivial if $a=0$. Assume $a\neq 0$. If $b=am$ and $a=bn$ then $a=amn$, so $mn=1$. Hence either $m=n=1$ or $m=n=-1$.

8.1. We have $a=cn$ and $b=cm$, hence $r=b-aq=c(m-nq)$.

8.1. We have $b = am$, $c = aq + r$ and $c = bt + s$. Hence $s = c - bt = (aq + r) - (am)t = (q - mt)a + r$. Since $0 \le r < a$, the remainder of the division $s : a$ is r.

8.1. (a) $a^2 - 1 = (a - 1)(a + 1)$. (b) $a^n - 1 = (a - 1)(a^{n-1} + \ldots + a + 1)$.

8.3 Factorization into primes

8.3. Yes, the number 2.

8.3. (a) p occurs in the prime factorization of ab, so it must occur in the prime factorization of a or in the prime factorization of b.

(b) $p|a(b/a)$, but $p \not| a$, so by (a), we must have $p|(b/a)$.

8.3. Let $n = p_1 p_2 \ldots p_k$; each $p_i \ge 2$, hence $n \ge 2^k$.

8.3. If $r_i = r_j$ then $ia - ja$ is divisible by p. But $ia - ja = (i - j)a$ and neither a nor $i - j$ are divisible by p. Hence the r_i are all different. None of them is 0. Their number is $p - 1$, so every value $1, 2, \ldots, p - 1$ must occur among the r_i.

8.3. For a prime p, the proof is the same as for 2. If n is composite but not a square, then there is a prime p that occurs in the prime factorization of n an odd number of times. We can repeat the proof by looking at this p.

8.3. Fact: If $\sqrt[k]{n}$ is not an integer then it is irrational. Proof: there is a prime that occurs in the prime factorization of n, say t times, where $k \not| t$. If (indirect assumption) $\sqrt[k]{n} = a/b$ then $nb^k = a^k$, and so the number of times p occurs in the prime factorization of the ;left hand side is not divisible by k, while the number of times it occurs in the prime factorization of the right hand side is divisible by k. A contradiction.

8.5 Fermat's "Little" Theorem

8.5. $4 \not| \binom{4}{2} = 6$. $4 \not| 2^4 - 2 - 14$.

8.5. (a) What we need that each of the p rotated copies of a set are different. Suppose that there is a set which occurs a times. Then trivially every other set occurs a times. But then $a|p$, so we must have $a = 1$ or p. If all p rotated copies are the same then trivially either $k = 0$ or $k = p$, which were excluded. So we have $a = 1$ as claimed. (b) Consider the set of two opposite vertices of a square. (c) If each box contains p subsets of size k, the total number of subsets must be divisible by k.

8.5. We consider each number to have p digits, by adding zeros at the front if necessary. We get p numbers from each number a by cyclic shift. These are all the same when all digits of a are the same, but all different otherwise (why? the assumption that p is a prime is needed here!). So we get $a^p - a$ numbers that are divided into classes of size p. Thus $p|a^p - a$.

8.5. Assume that $\gcd(a, p) = 1$. Consider the product $a(2a)(3a)\ldots((p - 1)a) = (p - 1)!a^{p-1}$. Let r_i be the remainder of ia when divided by p. Then the product above has the same remainder when divided by p as the product $r_1 r_2 \ldots r_{p-1}$. But this product is just $(p - 1)!$. Hence p is a divisor of $(p-1)!a^{p-1} - (p-1)! = (p-1)!(a^{p-1} - 1)$. Since p is a prime, it is not a divisor of $(p - 1)!$, and so it is a divisor of $a^{p-1} - 1$.

8.6 The Euclidean Algorithm

8.6. $\gcd(a, b) \le a$, but a is a comon divisor, so $\gcd(a, b) = a$.

8.6. Let $d = \gcd(a, b)$. Then $d|a$ and $d|b$, and hence $d|b - a$. Thus d is a common divisor of a and $b - a$, and hence $\gcd(a, b) = d \le \gcd(a, b)$. A similar argument show the reverse inequality.

8.6. (a) $\gcd(a/2, b)|(a/2)$ and hence $\gcd(a/2, b)|a$. So $\gcd(a/2, b)$ is a common divisor of a and b and hence $\gcd(a/2, b) \le \gcd(a, b)$. The reverse inequality follows similarly, using that $\gcd(a, b)$ is odd, and hence $\gcd(a, b)|(a/2)$.

(b) $\gcd(a/2, b/2)|(a/2)$ and hence $2\gcd(a/2, b/2)|a$. Similary, $2\gcd(a/2, b/2)|b$, and hence $2\gcd(a/2, b/2) \le \gcd(a, b)$. Conversely, $\gcd(a, b)|a$ and hence $\frac{1}{2}\gcd(a, b)|a/2$. Similary, $\frac{1}{2}\gcd(a, b)|b/2$, and hence $\frac{1}{2}\gcd(a, b) \le \gcd(a/2, b/2)$.

8.6. Consider each prime that occurs in either one of them, raise it to the larger of the two exponents, and multiply these prime powers.

8.6. If a and b are the two integers, and you know the prime factorization of a, then take the prime factors of a one by one, divide b with them repeatedly to determine their exponent in the prime factorization of b, raise them to the smaller of their exponent in the prime factorizations of a and b, and multiply these prime powers.

8.6. By the descriptions of the gcd and lcm above, each prime occurs the same number of times in the prime factorization of both sides.

8.6. $\gcd(a, a + 1) = \gcd(a, 1) = \gcd(0, 1) = 1$.

8.6. The remainder of F_{n+1} divided by F_n is F_{n-1}. Hence $\gcd(F_{n+1}, F_n) = \gcd(F_n, F_{n-1}) = \ldots = \gcd(F_3, F_2) = 1$. This lasts $n - 1$ steps.

8.6. By induction on k. True if $k = 1$. Suppose that $k > 1$. Let $b = aq + r$, $1 \le r < a$. Then the euclidean algorith for computing $\gcd(a, r)$ lasts $k - 1$ steps, hence $a \ge F_k$ and $r \ge F_{k-1}$ by the induction hypothesis. But then $b = aq + r \ge a + r \ge F_k + F_{k-1} = F_{k+1}$.

8.6. (a) Takes 10 steps. (b) Follows from $\gcd(a, b) = \gcd(a - b, b)$. (c) $\gcd(10^{100} - 1, 10^{100} - 2)$ takes $10^{100} - 1$ steps.

8.6. (a) Takes 8 steps. (b) At least one of the numbers remains odd all the time. (c) Follows from exercises 8.6 and 8.6. (d) The product of the two numbers drops by a factor of two in one of any two iterations.

8.7 Testing for primality

8.7. By induction on k. True if $k = 1$. Let $n = 2m + a$, where a is 0 or 1. Then m has $k - 1$ bits, so by induction, we can compute 2^m using at most $2(k - 1)$ multiplications. Now $2^n = (2^m)^2$ if $a = 0$ and $2^n = (2^m)^2 \cdot 2$ if $a = 1$.

8.7. If $3|a$ then clearly $3|a^{561} - a$. If $3 \not| a$, then $3|a^2 - 1$ by Fermat, hence $3|(a^2)^{280} - 1 = a^{560} - 1$. Similarly, if $11 \not| a$, then $11|a^{10} - 1$ and hence $11|(a^{10})^{56} - 1 = a^{560} - 1$. Finally, if $17 \not| a$, then $17|a^{16} - 1$ and hence $17|(a^{16})^{35} - 1 = a^{560} - 1$.

9 Graphs

9.1 Even and odd degrees

9.1. There are 2 graphs on 2 nodes, 8 graphs on 3 nodes (but only four "essentially different"), 16 graphs on 4 nodes (but only 11 "essentially different").

9.1. (a) No; sum of degrees must be even. (b) No; node with degree 5 must be connected to all other nodes, so we cannot have a node with degree 0. (c) 12. (d) $9 \cdot 7 \cdot 5 \cdot 3 \cdot 1 = 945$.

9.1. This graph, the *complete graph* has $\binom{n}{2}$ edges if it has n nodes.

9.1. (a) a path with 3 nodes; (b) a star with 4 endpoints; (c) the union of two paths with 3 nodes.

9.1. In graph (a), the number of edges is 17, the degrees are $9, 5, 3, 3, 2, 3, 1, 3, 2, 3$. In graph (b), the number of edges is 31, the degrees are $9, 5, 7, 5, 8, 3, 9, 5, 7, 4$.

9.1. $\binom{10}{2} = 45$.

9.1. $2^{\binom{20}{2}} = 2^{190}$.

9.1. *Every graph has two nodes with the same degree.* Since each degree is between 0 and $n-1$, if all degrees were different then they would be $0, 1, 2, 3, \dots n-1$ (in some order). But the node woth degree $n-1$ must be connected to all the others, in prticular to the node with degree 0, which is impossible.

9.2 Paths, cycles, and connectivity

9.2. There are 8 such graphs.

9.2. The empty graph on n nodes has 2^n subsgraphs. The triangle has 18 subgraphs.

9.2. Yes, the proof remains valid.

9.2 (a) Delete any edge from a path. (b) Consider two nodes u and v. the original graph contains a path connecting them. If this does not go through e, the it remains a path after e is deleted. If it goes through e, then let $e = xy$, and assume that the path reaches x first (when traversed from u to v). Then in the graph after e is deleted, there is a path from u to x, and also from x to y (the remainder of the cycle), so there is one from u to y. but there is also one from y to v, so there is also a path fom u to v.

9.2. (a) Consider a shortest walk from u to v; if this goes through any nodes more than once, the part of it between two passes through this node can be deleted, to make it shorter. (b) The two paths together form a walk from a to c.

9.2. Let w be a common node of H_1 and H_2. If you want a path between nodes u and v in H, then we can take a path from u to w, followed by a path from w to v, to get a walk from u to w.

9.2. Both graphs are connected.

9.2. The union of this edge and one of these components would form a connected graph that is strictly larger than the component, contradicting the definition of a component.

9.2. If u and v are in the same connected component, then this component, and hence G too, contains a path connecting them. Conversely, if there is a path P in G connecting u and v, then this path is a connected subgraph, and a maximal connected subgraph containing P is a connected component containing u and v.

9.2. Assume that the graph is not connected and let a connected component H of it have k nodes. Then H has at most $\binom{k}{2}$ edges. The rest of the graph has at most $\binom{n-k}{2}$ edges. Then the number of edges is at most

$$\binom{k}{2} + \binom{n-k}{2} = \binom{n-1}{2} - \frac{(k-1)(n-k-1)}{2} \le \binom{n-1}{2}.$$

10 Trees

10. If G is a tree than it contains no cycles (by definition), but adding any new edge creates a cycle (with the path in the tree connecting the endpoints of the new edge). Conversely, if a graph has no cycles but adding any edge creates a cycle, then it is connected (two nodes u

and v are either connected by an edge, or else adding an edge connecting them creates a cycle, which contains a path between u and v in the old graph), and therefore it is a tree.

10. If u and v are in the same connected component, then the new edge uv forms a cycle with the path connecting u and v in the old graph. If joning u and v by a new edge creates a cycle, then the rest of this cycle is path between u and v, and hence u and v are in the same component.

10. Assume that G is a tree. Then there is at least one path between two nodes, by connectivity. But there cannot be two paths, since then we would get a cycle (find the node v when the two paths branch away, and follow the second path until it hits the first path again; follow the first path back to v, to get a cycle).

Conversely, assume that there is a unique path between each pair of nodes. Then the graph is connected (since there is a path) and cannot contain a cycle (since two nodes on the cycle would have at least two paths between them).

10.1 How to grow a tree?

10.1. Start the path from a node of degree 1.

10.1. Any edge has only one lord, since if there were two, they would have to start from different ends, and they would have then two ways to get to the King. Similarly, and edge with no lord would have to lead to two different ways.

10.1. Start at any node v. If one of the branches at this node contains more than half of all nodes, move along the edge leading to this branch. Repeat. You'll never backtrack because this would mean that there is an edge whose deletion results in two connected components, both containing more than half of the nodes. You'll never cycle back to a node already sen because the graph is a tree. Therefore you must get stuck at a node such that each branch at this node contains at most half of all nodes.

10.2 Rooted trees

10.3. The number of unlabelled trees on $2, 3, 4, 5$ nodes is $1, 1, 2, 3$. They give rise to a total of $1, 2, 16, 125$ labelled trees.

10.3. There are n stars and $n!/2$ paths on n nodes.

10.4 How to store a tree?

10.4. The first is the father code of a path; the third is the father code of a star. The other two are not father codes of trees.

10.4. This is the number of possible father codes.

10.4. Define a graph on $\{1, \ldots, n\}$ by connecting all pairs of nodes in the same column. If we do it backwards, starting with the last column, we get a procedure of growing a tree by adding new node and an edge connecting it to an old node.

10.4. (a) encodes a path; (b) encodes a star; (c) does not encode any tree (there are more 0's than 1's among the first 5 elements, which is impossible in the planar code of any tree).

11 Finding the optimum

11.1 Finding the best tree

11.1. Let H be an optimal tree and let G be the tree contructed by the pessimistic government. Look at the first step when an edge $e = uv$ of H is eliminated. Deleting e from H we get two components; since G is connected, it has an edge f connecting these two components. The edge f cannot be more expensive than e, else the pessimistic government would have chosen f to eliminate instead of e. But then we can replace e by f in H without increasing its cost. Hence we conclude as in the proof given above.

11.1. [Very similar.]

11.1. [Very similar.]

11.1. Take nodes $1, 2, 3, 4$ and costs $c(12) = c(23) = c(34) = c(41) = 3$, $c(13) = 4$, $c(24) = 1$. The pessimistic government builds (12341), while the best solution is 12431.

11.2 Traveling Salesman

11.2. No, because it intersects itself (see next exercise).

11.2. Replacing two intersecting edges by two other edges pairing up the same 4 nodes, just differently, gives a shorter tour by the triangle inequality.

12 Matchings in graphs

12.1 A dancing problem

12.1. If every degree is d, then the number of edges is $d \cdot |A|$, but also $d \cdot |B|$.

12.1. (a) A triangle; (b) a star.

12.1. A graph in which every node has degree 2 is the union of disjoint cycles. If the graph is bipartite, these cycles have even length.

12.3. Let $X \subseteq A$ and let Y denote the set of neighbors of X in B. There are exactly $d|X|$ edges starting from X. Every node in Y accomodates no more than d of these; hence $|Y| \geq |X|$.

12.4. On a path with 4 nodes, we may select the middle edge.

12.4. The edges in M must meet every edge in G, in particular every edge in the perfect matching matching. So every edge in the perfect matching has at most one endpoint unmatched by M.

12.4. The largest matching has 5 edges.

12.4. If the algorithm terminates without a perfect matching, then the set S shows that the graph is not "good".

12.5. The first graph does; the second does not.